Comparative Study of Voltage-gated Potassium Channels Using Machine Learning

By

Bin Li

A thesis submitted to the Faculty of Graduate Studies and Research in partial Fulfillment of the requirements for the degree of

Doctor of Philosophy

Department of Biological Sciences
Edmonton, Alberta
Spring, 2005

Library and Archives Canada

Published Heritage Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et Archives Canada

Direction du Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

0-494-08263-1

Your file *Votre référence*
ISBN
Our file *Notre reference*
ISBN

NOTICE:
The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:
L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

This thesis is dedicated to my grandparents for their humble, honest, hardworking, and graceful lives.

Abstract

With the progress of genome projects and other high throughput applications, biological data have been growing exponentially. Consequently, data management and data mining have become indispensable components of biology. Computational analyses are being widely used in not only large genomics and proteomics projects, but also in studies of individual protein families. In this project, I took an "*in silico*" approach to collect, manage and explore the structural and functional data of voltage-gated potassium channels (VKCs). VKCs sense change in transmembrane voltage and open to allow potassium ions to pass through an ion-selective pore. They play critical roles in electrically excitable cells. Dysfunctional VKCs are related to diseases including epilepsy and cardiac arrhythmia. I first collected biological information on available VKCs from GenBank, Swissprot, and journal articles. These data and related analyses were stored in a relational database, called the voltage-gated potassium channel database (VKCDB http://vkcdb.biology.ualberta.ca). Using the collected data, I then built a predictor using a k-nearest neighbor classifier and feature selection techniques. The predictor successfully predicts the voltage sensitivity of a VKC based on its amino acid sequence with a mean absolute error of 7.0mV, and has been validated by permutation tests and independent experimental data. During the learning process, a number of residues were identified as being critical structural elements for modulating voltage sensitivity of VKCs. The methods I used in constructing VKCDB and building the computational model are not specifically tailored for VKCs; they can be easily generalized for study of other protein families.

Acknowledgements

I would like to first thank my supervisor, Dr. Warren Gallin, for trusting me with a project that I did not have much background in and teaching me everything I know about scientific research throughout these years. As much as I want to finish my study, I wish Warren could be my supervisor again. I also want to thank Drs. David Wishart and Russ Greiner for all the meetings, reference letters, and, more importantly, their expertise that has been continuously implementing my research.

My gratitude also goes to Lorne Leclair for his help in configuring and maintaining servers for VKCDB and other research I have worked on, Donna for making the lab so much nicer a place, and everybody in Warren's lab for their generous help. I'd specially like to thank my wife, Haiyan Zhang, for her immediate help in programming on countless occasions, for her being here with me for so many years, and for my two lovely sons.

Table of Contents

Chapter 1: Introduction 1

I. Voltage-gated potassium channels 1

1. Potassium channels are a family of integral membrane proteins 1
2. Potassium channels are broadly diversified 2
3. Voltage-gated potassium channels 3
4. Voltage-gated potassium channel is a tetramer 5
5. K^+ selectivity and voltage sensing of VKCs 6
6. Modulating voltage sensitivity of VKCs 14

II. Machine learning 16

1. Biological information explosion 16
2. Training data 17
3. Learning algorithms 19
   3.1 Decision Tree 19
   3.2 Naïve Bayes classifier 21
   3.3 Kernel density classifier 23
   3.4 K nearest neighbor classifier 23
   3.5 OneR classifier 26
4. Feature selection 26
5. Evaluation of learning 28

III. Thesis outline 32

Chapter 2: VKCDB: Voltage-gated potassium channel database 33

I. Introduction 33

II. Construction and content 34

III. Utility and discussion 39

IV. Conclusions.....41

V. Availability.....42

Chapter 3: Computational analysis of voltage-gated potassium channels 43

I. Introduction.....43

II. Methods.....47

1. Datasets.....47
2. Problem formulation.....48
3. Basic learning algorithms.....48
4. Feature selection.....50
5. Residue swapping.....51
6. Distance matrices in k nearest neighbor classification (KNN).....51
7. Outlier selection.....52
8. Final predictor construction.....53

III. Results.....53

1. Learning without feature selection.....53
2. Learning with a filter algorithm 56
3. Learning with a wrapper algorithm 56
4. Learning combined with outlier selection.....57
5. Identification of functionally critical residues.....60

IV. Discussion.....60

1. Learning with high dimensional data.....60
2. Outlier selection.....64
3. Identification of biologically important residues (features).....66

V. Conclusions.....68

Chapter 4: Validation of computational analysis 70

I. Introduction.....70

II. Methods.....72

1. Computational model and dataset.....72
2. Permutation test I.....72
3. Permutation test II.....72
4. Test datasets with experimental data.....73

III. Results.....73

1. Permutation tests.....73
2. Evaluation with VKC wild type data.....74
3. Evaluation with VKC mutant data.....76

IV. Discussion.....76

1. Statistical evaluation using permutation tests.....76
2. Evaluation of predictor using independent experimental data.....79

V. Conclusions.....84

Chapter 5: General discussion and conclusions 85

I. Biological data collection and management.....85

1. Information explosion.....85
2. Protein family databases.....87
3. A comprehensive resource for VKC research.....89

II. Computational analysis of biological data.....91

1. Machine learning can help.....91
2. Quality and quantity of training data.....93
3. Enrichment of training data.....97
4. Learning algorithms.....98

5. Feature selection.....99
6. Evaluation of learning.....101
7. Biological significance of selected residues.....102

III. Conclusions.....106

References.....109

List of Tables

Table 3.1: Best feature sets 58

Table 4.1: Permutation tests 75

Table 4.2: Validation with wild type VKC test data 75

Table 4.3: Validation with VKC mutant data 77

Table 4.4: Variations of V_{50} values in different cells 82

Table 5.1: Residue types of selected features 103

List of Figures

Figure 1.1: G-V curve of a voltage-gated potassium channel 4

Figure 1.2: Voltage-gated potassium channels 7

Figure 1.3: Signature motif of potassium channel 9

Figure 1.4: Ion selectivity of potassium channels 10

Figure 1.5: Two models of voltage-gating 12

Figure 1.6: Decision Tree classification 20

Figure 1.7: k-nearest neighbor classification 24

Figure 1.8: Forward selection 29

Figure 1.9: Heuristic search in a wrapper algorithm 30

Figure 2.1: Populating VKCDB 35

Figure 2.2: ER model of VKCDB 37

Figure 2.3: Screen dump of a VKCDB entry 40

Figure 3.1: Problem formulation 49

Figure 3.2: Construction of V_{50} predictor 54

Figure 3.3: Learning performances 55

Figure 3.4: Outlier selection 59

Figure 3.5: Informative features 61

Figure 4.1: Distance tree of training and test data 81

Figure 5.1: Growth of GenBank 86

Figure 5.2: Distance tree of training data 95

Figure 5.3: Selected residues mapped onto VKC structures 104

Figure 5.4: Selected residues and neighboring charged residues 107

Abbreviations

Ala	Alanine
CGI	Common gateway interface
CHO cell	Chinese hamster ovary cell
ER	Entity-Relationship
Fab	Antigen-binding fragment
G-V curve	Conductance-voltage curve
HEK cell	Human embryonic kidney cell
KCNQ	Potassium voltage-gated channel, KQT-like subfamily
KNN	K nearest neighbor
Kv subfamily	Voltage-gated potassium channel subfamily
KvAP	A voltage-gated potassium channel from *Aeropyrum pernix*
MAE	Mean absolute error
Neuro-2a cell	Neuroblastoma cell
PCA	Principal component analysis
PDB	Protein data bank
RBL cell	Rat basophilic leukemia cell
RMSD	Square root of mean square deviations
SD	Standard deviation
SNP	Single nucleotide polymorphism
SVM	Support vector machine
V_{50}	Half activation voltage
VC	Vapnik-Chervonenkis
VKC	Voltage-gated potassium channel
VKCDB	Voltage-gated potassium channel database
XML	Extensible Markup Language

Chapter 1: Introduction

I. Voltage-gated potassium channels

1. Potassium channels are a family of integral membrane proteins

As an autonomous functional entity of living organisms, the cell is separated from its surrounding environment by a highly specialized structure, the plasma membrane. The cell membrane not only forms a barrier to separate the cell's contents from the surrounding environment, it is also organized to permit communication with other cells and the surrounding environment, and to regulate selective transportation of different molecules in and out of the cell (Alberts et al., 2002). To maintain osmotic balance, the concentrations of ions and other molecules inside and outside of the cell require tight control and maintenance by the cell membrane. However, many cellular processes, for example, the electrical signaling of neurons, need a fast flow of ions across the membrane (Hille, 2001).

In 1950s, Hodgkin and Huxley published a series of papers on studies of the action potential in the squid giant axon (Hodgkin and Huxley, 1952a, b, c, d). They concluded that, for the action potential to occur, the permeability of the axon membrane to potassium and sodium ions has to undergo changes during the process (Hodgkin and Huxley, 1952b). Later, seminal work by Armstrong and Hille demonstrated that it is a unique protein pore, namely an ion channel, that allows K^+ or Na^+ ions to pass across cell membranes (Hille, 1970; Armstrong, 1981). Since the first ion channel was cloned (Noda et al., 1984), numerous functional studies with many ion channels have been done and have provided us with a flood of information on the structure and function of ion channels (Jan and Jan, 1997a; Wood and Baker, 2001; Sather and McCleskey, 2003).

Moreover, a series of milestone studies on ion channel structures by MacKinnon's lab and others in the past five years are finally bringing us a clear picture on how potassium channels selectively permit the passage of only K^+ ions at an extremely high efficiency (Doyle et al., 1998; Kreusch et al., 1998; Morais Cabral et al., 1998; Gulbis et al., 2000; Jiang et al., 2001; Sokolova et al., 2001; Jiang et al., 2002a, b; Jiang et al., 2003a; Kuo et al., 2003). The structural data also indicate several possible mechanisms by which different K^+ channels direct the opening and closing of the channel pore (Choe, 2002; Yellen, 2002; MacKinnon, 2004).

2. Potassium channels are highly diversified

Potassium channels are one of the most diverse protein families that we know (Jan and Jan, 1997a). They have been shown to exist in both prokaryotic and eukaryotic cells. Different potassium channels open and close in response to different cues (Jan and Jan, 1997a). In other words, the gating mechanisms are different. Ligand-gated potassium channels are activated to allow K^+ ions to pass when a specific ligand binds to a certain domain of the channel (Zagotta and Siegelbaum, 1996; Wollmuth and Sobolevsky, 2004). The probability that voltage-gated potassium channels will open begins to become significant when the difference in voltage across the cell membrane reaches a certain threshold (Yellen, 2002). Some channels, for example, the Ca^{2+} activated potassium channel, react to both cytoplasmic factors and voltage differences (Wu, 2003). The bacterial potassium channel KcsA from *Streptomyces lividans* is gated by the intracellular pH (Schrempf et al., 1995). Other potassium channels, including some inward rectifiers, are intrinsically open and are an important component in setting the resting potential of a cell (Stanfield et al., 2002).

Potassium channels play significant roles in maintaining membrane potential and shaping the action potential and firing patterns of excitable cells (Jan and Jan, 1997a), which are central to numerous physiological processes. Many neurological and cardiac disorders are associated with dysfunctional potassium channels (Ashcroft, 2000). A number of mutant potassium channel genes are indirectly related to, or directly cause, a number of diseases, including cardiac arrhythmia (Jentsch, 2000), epilepsy (Lerche et al., 2001), diabetes (Smits, 1996), and cancers (Abdul and Hoosein, 2002b). The present research is primarily focused on one group of potassium channels, the voltage-gated potassium channels.

3. Voltage-gated potassium channels

Typically, the resting membrane potential, the difference in voltage between inside and outside of an electrically unexcited cell, is negative (Hille, 2001). In other words, the cell membrane is polarized. For example, the resting potentials of human and frog cells are normally around –90mV and –80mV, respectively (Hille, 2001).

A voltage-gated potassium channel (VKC) opens and closes in response to change in transmembrane potential (Yellen, 2002). For electrically excitable cells, such as neurons, some physiological events, usually excitatory input from neighboring neurons, can shift the potential inside the cell toward a more positive direction and thus depolarize the cell membrane. When the transmembrane potential is depolarized past a characteristic threshold, the possibility that a VKC is activated and opens to permit K^+ ion flow will increase significantly (Figure 1.1). When a VKC opens, K^+ ions can pass through at a speed close to free flow, yet with a remarkably strict selectivity that favors K^+ ions many times over smaller monovalent Na^+ ions (Yellen, 2002). The molecular mechanisms by

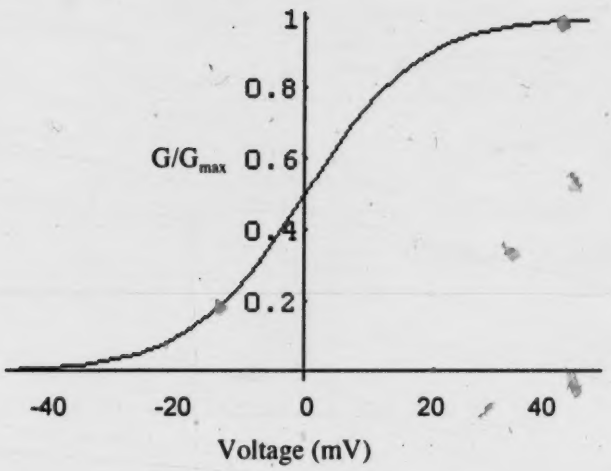


Figure 1.1: G-V curve of a voltage-gated potassium channel (VKC). At the resting potential of –90mV in human cells, this VKC remains closed. Indicated by the conductance (G), the probability that this VKC will open begins to become significant at around –40mV. The half activation voltage (V_{50}), at which 50% of the channels open, is 0mV, yielding 50% of the maximal conductance (G_{max}).

which voltage gating and potassium selectivity are achieved have been under study for decades.

4. Voltage-gated potassium channel is a tetramer

A typical voltage-gated potassium channel (VKC) gene encodes a membrane protein of 400 ~ 600 amino acids (Tempel et al., 1987; Stuhmer et al., 1989). Computational predictions suggest there are six transmembrane domains (S1 – S6) in each VKC gene product (Tempel et al., 1987; Stuhmer et al., 1989). This prediction is generally supported by various functional analyses (Monks et al., 1999; Hong and Miller, 2000; Li-Smerin et al., 2000) and recent structural data (Jiang et al., 2003a).

In the early 1990's, it was not clear if a VKC functions as a monomer or multimer even though a number of VKC genes had been cloned. Mackinnon first solved this mystery using simple binomial statistics (MacKinnon, 1991a). If a VKC is a multimer and the binding of each subunit encoded by a single VKC gene is an independent process, similar to the event of flipping coins, the probability (P) of the presence of a specific subunit in a VKC complex will follow the binomial distribution. For example, if A and B are two subunits that can combine to form a functional VKC, P(A) and P(B) are fractions of each subunit in the cell, the probability of observing n subunits in the VKC complex if VKCs are an N-mer can be calculated by:

$$P = N!P(A)^{n}P(B)^{N-n}/n!(N-n)! \qquad \text{Formula 1.1}$$

Mackinnon used a wild type VKC subunit (P(A) = 0.1) and a mutant (P(B) = 0.9) that essentially abolishes the binding of scorpion toxin only when all subunits of a VKC

complex are the mutant subunit. In other words, one or more wild type subunits can rescue the toxin sensitivity of a VKC complex. Using Formula 1.1, it can be calculated that the probability of observing a mutant homomer that is insensitive to the toxin is 0.9, 0.81, 0.729, 0.656 or 0.59, if a functional VKC comprises of one, two, three, four or five subunits, respectively. A value of 0.65 was observed in the experiment, which suggested that a VKC is a tetramer (MacKinnon, 1991a). Together with evidence from other studies (Isacoff et al., 1990; Liman et al., 1992), it became evident that a functional VKC consists of four protein subunits (Figure 1.2A).

5. K^+ selectivity and voltage sensing of voltage-gated potassium channels

Extensive electrophysiological and pharmacological studies, and, more recently, structural determinations of wild type VKCs and VKC mutants have been carried out to identify critical structural elements and characterize the structure-functional relationship of VKCs (Sigworth, 1994; Yellen, 1998; Bezanilla, 2000; Choe, 2002). Two major components of VKC functioning are K^+ selectivity and voltage sensing.

The last two transmembrane domains, S5-S6, including a loop between them (Figure 1.2B) have often been targeted in structure-functional studies of VKCs. This is because a number of mutants in this region drastically altered K^+ selectivity, conductance, and toxicological features of the channel (MacKinnon and Yellen, 1990; Hartmann et al., 1991; Heginbotham and MacKinnon, 1992; Yellen, 2001). In particular, mutations within a successive five-residue fragment, TVGYG, significantly change or even abolish the K^+ selectivity and conductance (Heginbotham et al., 1992; Heginbotham et al., 1994). This motif is the K^+ specific selective filter and was later called "the

Figure 1.2: Voltage-gated potassium channels.

A: There are four similar subunits in a functional voltage-gated potassium channel, and they are shown here in different colors. This is the intracellular view of a voltage-gated potassium channels (Jiang et al., 2003a). © Nature Publishing Group

B: There are six transmembrane domains in each subunit. It is generally accepted that S4 is the voltage sensor with a characteristic charged residue at every third position; S5, S6 and the linking loop make up the channel pore; S1-S3 are likely to modulate the voltage sensitivity of the channel. T1 is the tetramerization domain.

signature sequence" of potassium channels (Heginbotham et al., 1994). This became obvious when more voltage-gated potassium channels and other potassium channels from different organisms were cloned and sequenced. This signature sequence is the most conserved motif across all sequenced potassium channels (Figure 1.3).

The most convincing proof of this came from the first crystal structure of a bacterial potassium channel KcsA (Doyle et al., 1998). This bacterial potassium channel has two transmembrane domains and a linking loop, homologous to the S5-S6 helices in VKCs, which make up the actual channel pore. The structure of KcsA from *Streptomyces lividans* shows an inverted teepee conformation of the channel pore (Doyle et al., 1998). The signature motifs of four potassium channel subunits, TVGYG, line the narrowest part of the pore, which is the selective filter. The carbonyl oxygen atoms of these signature residues from four subunits form four continuous cubic cages that coordinate potassium ions at the center (Figure 1.4). The physical size of the cage fits snugly with potassium ions and discriminates against the smaller sodium ions (Doyle et al., 1998). A recent molecular dynamics simulation study indicates the selective filtering of potassium ion relies on electrostatic interactions and that potassium ion selectivity is not a consequence of a rigid structural fit but a flexible and dynamic process (Noskov et al., 2004).

However, the picture of voltage sensing and the coupling between voltage sensing and channel opening is far from clear. The voltage-sensing component was first proposed when the first voltage-gated ion channel was cloned (Noda et al., 1984). It is a voltage-gated sodium channel from *Electrophorus electricus,* consisting of four repeated homologous units. Each homologous unit is equivalent to one subunit in a VKC. In each unit, the amino acid sequence revealed that there were positively charged Arg or Lys

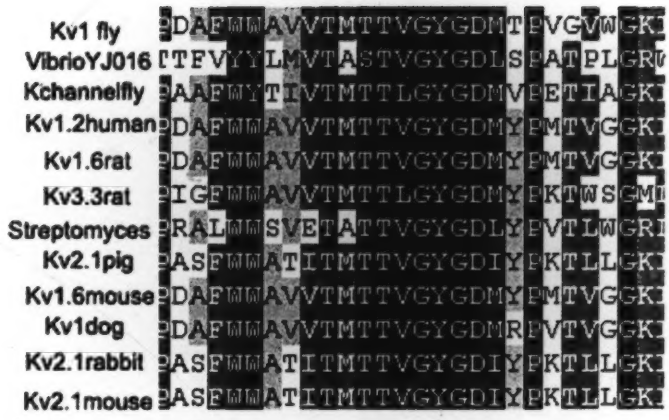


Figure 1.3: The signature motif of potassium channels. The signature motif, TVGYG, is the most conserved motif in potassium channels, as revealed here by a multiple sequence alignment of twelve potassium channels from a wide range of species.

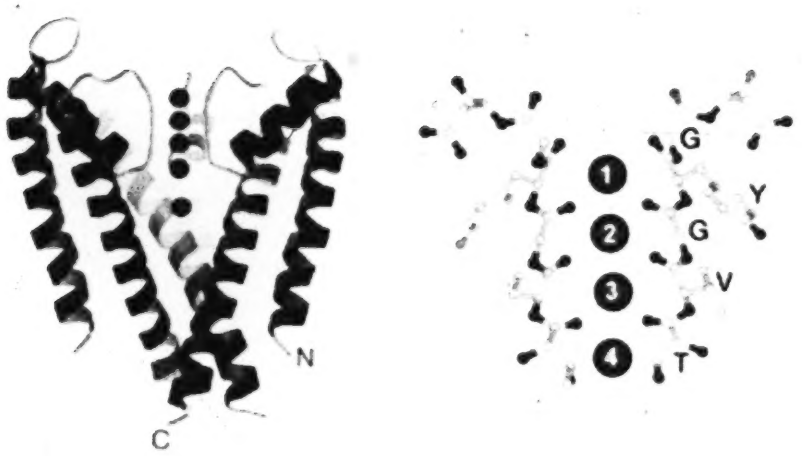


Figure 1.4: Potassium selectivity of KcsA was determined by the signature motif (Chung and Kuyucak (2002) Ion channels: recent progress and prospects. Eur Biophys. J. 31: 283-93, Figure 1. (Chung and Kuyucak, 2002). ©Springer. The carbonyl oxygen atoms of the signature residues from four subunits coordinate the passage of K^+ ions and discriminate against other molecules through the channel pore. Only two subunits are shown here. Green circles represent K^+ ions.

residues at every third position in the S4 domain (Noda et al., 1984; Tempel et al., 1987; Stuhmer et al., 1989). The unusual presence of repeated charged residues within a transmembrane domain immediately made the S4 domain a strong candidate for voltage sensing (Figure 1.2B). A number of subsequent studies with mutations at these charged positions were done, and they all reported altered channel opening properties (Liman et al., 1991; Papazian et al., 1991). Furthermore, accessibility studies using cysteine substitution and more recent fluorescence resonance energy transfer experiments showed that S4 moves during channel opening and several S4 residues even appear to move fully across the membrane (Yang and Horn, 1995; Yang et al., 1996; Cha and Bezanilla, 1997; Starace et al., 1997; Mannuzzu and Isacoff, 2000).

A voltage-sensing model was thus proposed in which, when the membrane potential is depolarized, the altered electric field forces the charge-carrying S4 domain to slide through the membrane (Catterall, 1986). The movement was described as a relatively small "screw-like" turn to accommodate the difficulty of moving within the hydrophobic environment of the cell membrane (Figure 1.5A).

Recently, the crystal structure of KvAP, a voltage-gated potassium channel from the thermophilic archea *Aeropyrum pernix* with all six transmembrane domains, was published (Jiang et al., 2003a). Based on the structure and a functional analysis, the authors proposed a new "paddle" model for voltage sensing (Jiang et al., 2003a; Jiang et al., 2003b). In this structure, S3 was found to take a helix-turn-helix conformation and S4 was located at the periphery of the protein. Upon activation, the second helix of S3 and S4 were proposed to "paddle" through the membrane and lead to the movement of S5 and channel opening (Figure 1.5B) (Jiang et al., 2003a; Jiang et al., 2003b).

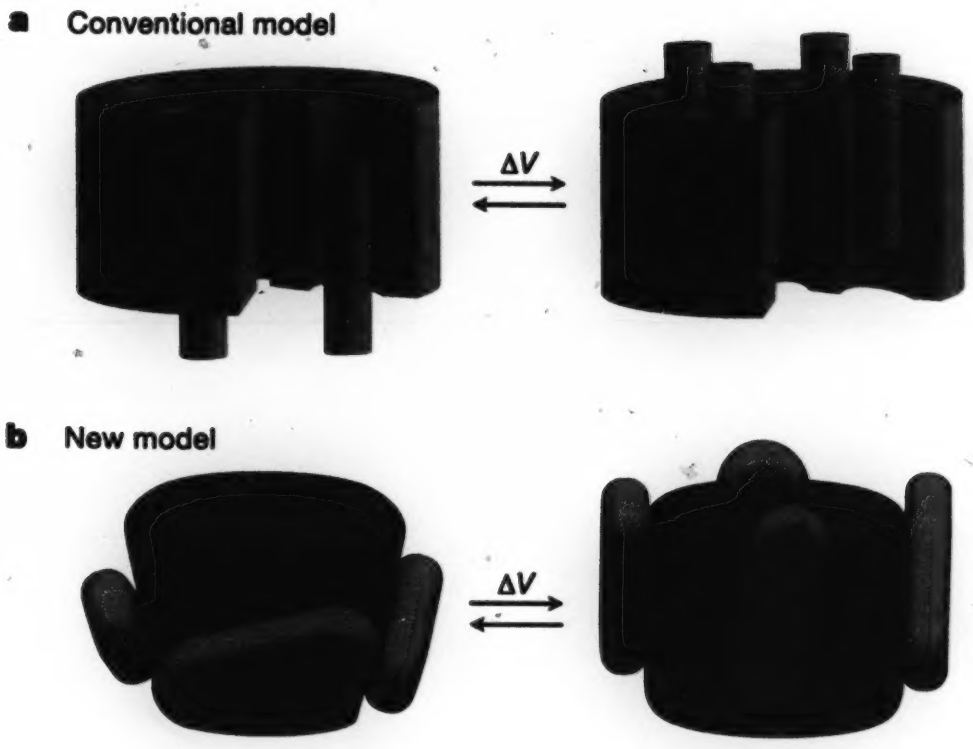


Figure 1.5: Two models for the movement of S4 during activation (Jiang et al., 2003a).

© Nature Publishing Group.

A: The traditional model positions S4 surrounded by other helices. S4 carries the gating charges along the membrane by translocation and rotation, and opens the channel pore.

B: The paddle model positions S4 at the protein/lipid interface. S4 is a part of the "sensor paddle" that moves across against the lipid membrane, leading to the opening of the pore.

However, since the S1-S4 structure was determined when it was stabilized by a bound Fab fragment, the structure is likely to have been distorted to some extent and may not reflect the native conformation of the channel (Laine et al., 2003; Laine et al., 2004). A number of experiments suggested that the interaction between negatively charged residues in S2 (E283 in *Shaker*) and positively charged residues in S4 (R368 and R371 in *Shaker*) plays a critical role in VKC activation (Tiwari-Woodruff et al., 1997). These interactions suggest that S4 is in the vicinity of the S2 helix (Laine et al., 2004). The formation of disulfide bonds and salt bridges between S4 and the pore domain in recent mutagenesis studies also indicated that S4 is likely located close to the pore domain (Laine et al., 2003), contrary to the KvAP structure (Jiang et al., 2003a). Additionally, the gating mechanism of an archeal VKC could be unique, compared with eukaryotic VKCs. Further study is still needed to clarify the detailed mechanism of voltage sensing and the coupling that leads to the channel opening.

For most bacterial potassium channels, each subunit contains only two transmembrane domains (Koprowski and Kubalski, 2001). These two domains are equivalent to the S5-S6 region in a VKC. A chimeric VKC with its S5-S6 region substituted by a two-domain bacterial potassium channel was shown to be properly synthesized, assembled, and expressed but failed to behave in a voltage-gated fashion (Caprini et al., 2001). It is generally accepted that the voltage sensing and gating of VKCs are mediated by two relatively independent modules (Nelson et al., 1999; Patten et al., 1999), and that the voltage-gated response relies on a precise coupling mechanism (Caprini et al., 2001). Generally, the S1-S4 region is considered the voltage sensing module, in which S4 is the main sensing unit and S1-S3 is thought to play an assisting

and modulating role, while S5-S6 is indisputably the gating module that directly controls the passage of K^+ ions.

6. Modulating voltage sensitivity of VKCs

VKCs share a high degree of sequence similarity. The most conserved regions and residues are critical in VKC functioning. Many mutations at these conserved positions directly interfere with voltage sensing and gating of VKCs and often lead to qualitative phenotypic changes (Liman et al., 1991; MacKinnon, 1991c; Papazian et al., 1991; Heginbotham et al., 1994). Despite the fact that VKCs are highly conserved and share the same basic functions, such as voltage sensing and K^+ ion permeability, the quantitative functional features can be drastically different among different VKCs. The half activation voltage (V_{50}) is one of the central electrophysiological parameters of VKCs that define the voltage sensitivity of VKCs. V_{50} is the transmembrane voltage at which 50% of the VKCs will open, as shown in Figure 1.1. The known V_{50} values for VKCs range from –40mV to +70mV (Chapter 2) (Li and Gallin, 2004).

Evidently, the structural basis for this diversity in voltage sensitivity resides in those varying residues. During evolution, many functionally critical residues remain unchanged because of selective pressure. For those residues that vary, some are selectively neutral. They are thus fixed by chance (neutral drift) and are typically not functionally relevant. However, variations in some residues do not arise by random drift; they are selectively favored because they generate functional diversity to yield improved phenotypes. It is thus valuable to identify those residues that play "accessory" roles to the core function of a protein but generate functional diversity. These variable residues act as "principals" in quantitatively modulating and fine-tuning specific functional aspects of

protein. This project is aimed at identifying structural elements (residues) of VKCs whose states (identities) are not critical in essential functionality but contribute to the quantitative diversity in the voltage sensitivity of VKCs.

Most mutagenesis and other structure-functional studies of VKCs have focused on evolutionarily more conserved residues, which are residues that are vital in voltage sensing or channel gating (Liman et al., 1991; MacKinnon, 1991c; Papazian et al., 1991; Heginbotham et al., 1994). For example, the unique positively charged residues at every third position in S4 have been obvious targets for mutagenesis. Papazian et al demonstrated that these charged residues are associated with gating to different extents, indicating a voltage sensing role for S4 (Papazian et al., 1991). Three negatively charged residues in S2 and S3 are highly conserved among all VKCs, and they have also been shown to be involved in channel activation. These residues may form salt bridges with the positive charges in S4 in one of the open or closed states (Papazian et al., 1995; Tiwari-Woodruff et al., 1997).

More recently, structural and functional analyses have shown that residues in the N-terminal tetramerization (T1) domain also display an ability to alter the half activation voltage (V_{50}) of VKCs. There are four major VKC subfamilies, Kv1-4. The tetramerization domain is responsible for tetramerization of four subunits from the same Kv subfamily (Chandy, 1991) to form a functional VKC (Shen et al., 1993). This alteration is potentially mediated by a conformational change communicated through the loop between T1 and S1 (Cushman et al., 2000; Minor et al., 2000) or by direct interaction with other cytoplasmic loops of the channel protein. The linkage between T1 and voltage sensitivity is somewhat unexpected due to the seemingly remote distance

between the T1 domain and the gating unit of VKCs. However, since the N terminal cytoplasmic domain of potassium channels has been shown to interact with several intracellular signaling molecules (Cachero et al., 1998; Tsai et al., 1999; Eldstrom et al., 2002), this function of T1 can potentially couple the gating of VKCs with cytoplasmic signaling and provide a missing link between VKC function and the intracellular environment.

Residues in other regions have also been linked to voltage sensitivity including the S4-S5 loop and the C terminal end of VKCs (Lu et al., 2002), as might be expected for such a highly cooperative process (Bezanilla, 2000).

To understand the dynamic relationship between different residues and the resulting variations in voltage sensitivity of VKCs, a typical approach has been to experimentally evaluate variations in voltage sensitivity of a limited number of VKC mutants (Monks et al., 1999; Li-Smerin et al., 2000; Minor et al., 2000; Yifrach and MacKinnon, 2002). The selection of mutants is normally biased by personal view of VKC functioning and is limited by available experimental resources. Evidently, the resulting conclusions are only based on and restricted by the specific VKC mutants being tested and the specific channels in which the mutations are tested. In the present study, we have applied machine learning techniques (Mitchell, 1997) to maximally and impartially utilize all available structural and functional data from VKCs, to help detect the relationship between sequence variations and diversity in voltage sensitivity of VKCs.

II. Machine learning

1. Biological information explosion

Computational tools have long been used in many branches of biology including phylogenetic analysis, quantitative genetics, and epidemiology. Since the human genome project was initiated and high throughput sequencing projects started generating an exponentially increasing amount of data (Collins et al., 2003; Benson et al., 2004), the biological information explosion has made computing technology an indispensable part of biology. Many databases, for example, GenBank (Benson et al., 2004), and computational tools, such as BLAST (Altschul et al., 1990), were developed to cope with the deluge of data.

Designed to handle large volumes of data, machine learning is beginning to thrive in its applications to biological data mining (Narayanan et al., 2002; Liu and Wong, 2003; Kapetanovic et al., 2004). In many genome centers or microarray facilities, machine learning is becoming a pivotal tool in their daily operations.

Machine learning identifies and generalizes a pattern from a given set of examples. While unsupervised learning, such as clustering, simply looks for similarity among available data without pre-defined classes, we will focus on classification learning. Classification learning is also called "supervised learning" because the actual class of each example is provided in the dataset and the learning is supervised by these classified instances to produce a model that fits these classifications. A supervised learning algorithm will thus take in a number of classified examples and attempt to extract a model from which future instances can be classified (Mitchell, 1997).

2. Training data

Machine learning relies on data examples, sometimes called a training set. In a training set, each example or instance is derived from an actual event that is to be

classified and predicted. Typically, instances are independent from each other in a training set.

An instance is defined by a fixed set of features or attributes. The value of a feature is a measurement of the quantity or identity of this feature. There are two main types of features: numerical features and nominal features. Numerical features are described with real numbers whose values have real arithmetic relationships. Nominal features are a set of predefined categorical descriptions of the feature. For example, the amino acid identities are nominal features if each residue in a protein sequence is considered as a feature. There may or may not be an arithmetic relation among the nominal descriptions for a specific feature. In the case of amino acid identity being a feature, some amino acid residues are evolutionarily closer to each other and different amino acids can be actually compared using an amino acid comparison matrix (Altschul, 1991; Henikoff and Henikoff, 1992).

In classification learning, each instance is labeled with a predetermined class. As with features, classes can also be nominal or numerical. Nominal classification handles instances labeled with a predefined set of categories. For instance, to diagnose if a patient has cancer based on his or her gene expression profile, the training instances are likely to be labeled with nominal classes: "Yes (Cancer)" and "No (Healthy)". Instead of discrete classes, numerical prediction classifies instances with continuous numerical values. For example, numerical prediction will be more appropriate if a quantitative functional parameter of a protein is to be predicted based on its amino acid sequence.

Taken together, a training set for classification learning can be seen as a matrix of instances that are characterized by a set of features and labeled with nominal categories

or numerical values. A training set is the sole source of information input for machine learning, so the quality of the training set is critical to learning.

3. Learning algorithms

A variety of learning algorithms have been developed to handle classification learning (Mitchell, 1997). The essence of all learning algorithms is to generalize a model that can best fit the data by searching through a large but finite hypothesis space. Here I introduce a number of learning algorithms that were used in the present study.

3.1 Decision Trees

Decision Trees are among the most intuitive approaches to learn from a set of independent instances. Starting from a root, a decision tree grows by adding one layer of nodes at a time. At each node, a certain evaluation function based on one or more features is applied and only the instances that pass the test can reach this node, otherwise they will be directed to other nodes. With a given training set, a decision tree is built in such a way that the final tree will correctly classify all training instances to different leaves, with each leaf representing each class. Most of the algorithms that are used to build a decision tree employ a greedy search to find the best tree from a hypothesis space of all possible decision trees (Quinlan, 1986).

When an "unknown" instance is to be classified, successive tests will be done at different layers of nodes, and the instance will move down the tree and eventually reach a leaf, where its class will be determined by the class to which the leaf is assigned. An example of classification using a decision tree is given in Figure 1.6.

Features	Goodenow	Bettman	Oilers	Fan
Instance 1	Sane	Sane	Survive	Happy

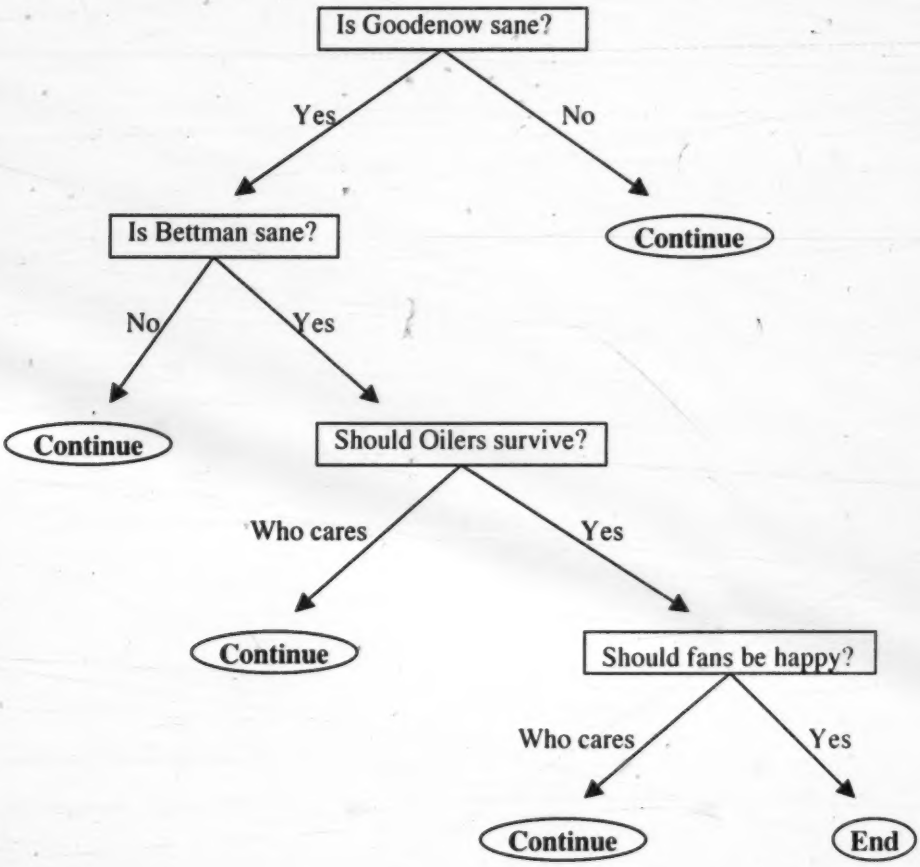


Figure 1.6: Decision Tree classification on whether the 2004-2005 NHL lockout will continue or end. The decision tree checks an instance at every node and allows the instance to follow the path only when it passes the test. When an instance gets to a leaf node, it will be assigned the class that leaf represents. Based on the given decision tree, the example (Instance 1) at the top will get to the "End" leaf, thus meaning that NHL lockout will end, believe it or not.

Decision Trees have been successfully used in many tasks (Masic et al., 1998; Selbig et al., 1999; Murphy, 2001; Viikki et al., 2002; Zorman et al., 2002; Jerez-Aragones et al., 2003; Kim, 2004a). The decision-making process of a decision tree is easy to understand and the relationship between the final classification and the related features can be readily extracted from a finished "tree". This "transparency" of decision tree makes it one of the more popular learning algorithms.

However, Decision Trees are susceptible to "overfitting", meaning the resulting classification works poorly on "unseen" instances despite an "excellent" performance with the training data (Mitchell, 1997). If there are a limited number of instances and the "tree" is allowed to grow deep enough to fit every instance, it tends to generate a model that overfits the training data. Thus, a shorter tree is always favored. A number of "pruning" methods are also available to alleviate this overfitting problem (Quinlan, 1986; Mingers, 1989; Quinlan, 1993).

3.2 Naïve Bayes classifier

Machine learning is closely related to statistical analysis. In fact, the famous statistical Bayes theorem is the cornerstone of many learning methods including the Naïve Bayes classifier. Bayes' theorem calculates the posterior probability of an event based on its prior probability and training instances. In Bayes' theorem:

$$P(B|A) = P(A|B)P(B)/P(A)$$

$P(B|A)$ is so-called the posterior probability of event B provided that event A occurs. Similarly, $P(A|B)$ means the probability that event A happens if event B exists. $P(A)$ and $P(B)$ denote the prior probability of event A and event B, respectively.

For a typical learning task, an unknown instance with a set of features (f_1, f_2, ..., f_n) is to be classified. In a Bayesian world, we can also rephrase the task as to find the most probable class (C_{max}) among all the possible classes (C_1, C_2, ..., C_m), given the set of features (f_1, f_2, ..., f_n). It can expressed as:

$$C_{max} = \max P(C_j | f_1, f_2, ..., f_n) \text{ for } (j = 1 \text{ to } m)$$

According to Bayes theorem:

$$C_{max} = \max P(f_1, f_2, ..., f_n | C_j) P(C_j) / P(f_1, f_2, ..., f_n)$$

Since $P(f_1, f_2, ..., f_n)$ does not depend on the class label, it needs not to be calculated. Thus:

$$C_{max} = \max P(f_1, f_2, ..., f_n | C_j) P(C_j)$$

$P(C_j)$ can be easily deduced by simply counting the frequency at which each class (C_j) occurs in the training data. $P(f_1, f_2, ..., f_n | C_j)$, however, is not feasible to obtain because it requires an unrealistic large number of instances to get a reliable estimate. The Naïve Bayes classifier, on the other hand, assumes that each feature occurs independently, therefore:

$$C_{max} = \max P(f_1, f_2, ..., f_n | C_j) P(C_j)$$
$$= \max P(f_1 | C_j) P(f_2 | C_j) \cdots P(f_n | C_j) P(C_j)$$

$P(f_i | C_j)$ can be estimated by counting the frequency at which each feature occurs for a given class (C_j) in the training set. Using this approach, an unknown instance can thus be classified as the most probable class (C_{max}).

Not only is Naïve Bayes classification simple, but in many cases, it outperforms other classification methods. In addition, Naïve Bayes classification explicitly handles and reports probabilities, a feature not found in most other learning algorithms. A potential problem with Naïve Bayes classification, however, resides in its assumption that

all features are independent of each other. If one or more pair of redundant or correlated features exists in the training data, the performance can suffer (Mitchell, 1997).

3.3 Kernel density classifier

As described in the previous section, the Naïve Bayes classifier obtains the posterior probability $P(f_i|C_j)$ by counting the occurrence of each feature for every class (C_j). However, this approach is only applicable for nominal features. If features are continuous numerical values, the Naïve Bayes classifier assumes the values follow a Gaussian distribution to estimate the probability. This assumption works well for data with features that indeed take a Gaussian distribution. However, for features with a different distribution, Naïve Bayes sometimes performs poorly.

The kernel density classifier is a variation to Naïve Bayes classification. To improve Naïve Bayes classification, researchers incorporated kernel density estimation, instead of assuming a Gaussian distribution. This implementation yields significantly better learning performances when training data with features that have a non-Gaussian distribution, and it generates comparable results with data that do have a Gaussian distribution (Herbrich, 2002).

3.4 k-nearest neighbor classifier

Instead of generating an explicit model from the training data, the nearest neighbor classifier simply stores all instances. When an unknown instance is to be classified, the nearest neighbor classifier compares the new instance with all the training data, and assigns the class label of most similar training example (neighbor) to the new instance (Mitchell, 1997). The k-nearest neighbor classifier assigns the average of the closest k neighbors to the new instance (Figure 1.7).

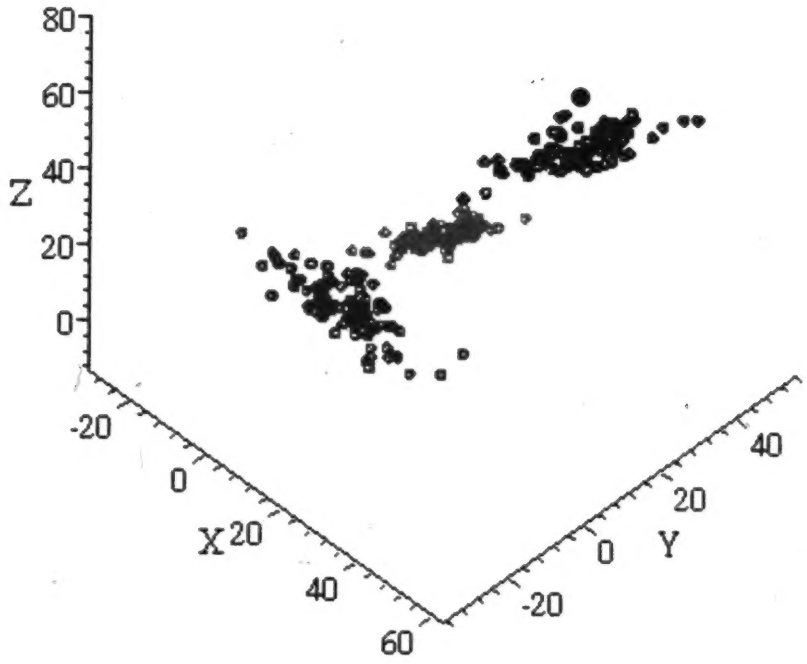


Figure 1.7: KNN (k-nearest neighbor) classification. A KNN classifier maps training data onto a multiple dimensional space, with each axis representing one feature. The KNN classifier then classifies a new instance based on its "nearest neighboring" class. As shown in this figure, training data are mapped onto a three dimensional space. The training data from the same class form a cluster (blue, green, and red). The new instance (black dot) will be classified as "blue" using KNN classification because its "nearest neighbor" is class "blue".

Typically, instances with n features can be mapped as points onto the n dimensional Euclidean space. The Euclidean distance between two points is used to measure the similarity or distance between the two instances. If instances are defined by n features (f_1, f_2, ..., f_n), the distance between two instances i and j is:

$$\text{Distance}(i, j) = \sqrt{\sum_{k=1}^{n} (f_{ik} - f_{jk})^2}$$

Since k-nearest neighbor classification does not generalize an explicit model, its performance is superior to other methods when the targeted event can not be described by a single data model. Many biological events are highly coordinated processes that involve multiple factors in a complicated fashion. It is often not possible to generalize the process using a simple mathematical function. Therefore, k-nearest neighbor classification has a number of advantages in biological data mining (Cabello et al., 1991; Yi and Lander, 1993; Cedeno and Agrafiotis, 2003; Jain and Mazumdar, 2003; Kim, 2004b).

As with other learning methods, the k-nearest neighbor classifier has its shortcomings. Because it utilizes all features to compute distances between training data, the existence of a large number of irrelevant features often causes problems. Although other methods share the same concern, k-nearest neighbor classification is especially sensitive to irrelevant features. In many cases, techniques such as feature selection are combined with k-nearest neighbor classifier to produce the best learning performance (Mitchell, 1997).

3.5 OneR classifier

OneR classifier selects the single feature that produces the best learning performance, and uses only this "best" feature for classification. Surprisingly, it has been

shown to obtain reasonable accuracy for some tasks (Holte, 1993). However, it is often primarily used to generate a baseline classification to evaluate other learning performance (Witten and Frank, 2000).

4. Feature selection

Machine learning generalizes from training data. Obviously, its performance relies on and, at the same time, is also limited by training data. In practice, the most common factor that limits learning performance is limited availability of training data. Since machine learning is achieved by searching through hypothesis space, the number of training data that are required to yield a good classification is often related to the complexity of the hypothesis space, which increases exponentially as the number of features goes up (Mitchell, 1997). The complexity of the hypothesis space can be evaluated by its Vapnik-Chervonenkis (VC) dimension, and the number of training instances required for successful learning can be estimated based on the VC dimension of the targeted hypothesis space (Blumer et al., 1989).

In many cases including the dataset in the present study, however, the number of features is prohibitively more than the number of available instances, which is sometimes called "the curse of dimension" (Mitchell, 1997). For learning with fewer than the "required" training examples, several feature selection techniques have been developed to lower the complexity of hypothesis space and lift the "curse" (Almuallim and Dietterich, 1991; Koller and Sahami, 1996; Blum and Langley, 1997; Kohavi and John, 1997; Yang and Pedersen, 1997).

Feature selection removes some features and retains only those features from the training data that are relevant to the learning task in order to improve the learning

performance. Individual learning algorithms, for example, Decision Trees (Quinlan, 1993), have been applied for the sole purpose of feature selection. Typically, a decision tree selects only a subset of features to evaluate instances in its final tree. These features can thus be considered relevant features and applied in other learning processes. Some statistical techniques, such as principal component analysis (Jolliffe, 2002), have also been used to compress the feature dimension. In machine learning, the filter and wrapper algorithms are two of the main approaches that use classification information for feature selection (Blum and Langley, 1997).

A filter algorithm evaluates all features based on training data, without taking into account the effects of learning algorithms. It preprocesses data by evaluating each feature using some standard (Blum and Langley, 1997). Some select the smallest subset of features that are sufficient for classifying training data (Almuallim and Dietterich, 1991); some rank features by their mutual information gain scores based on their association with the final classification (Koller and Sahami, 1996). The filter approach has successfully improved some learning performances, but for some datasets, it did not fare well (Blum and Langley, 1997).

Instead of relying on training data alone, the wrapper approach includes learning algorithms in its evaluation of features (Kohavi and John, 1997). Its name comes from the fact that a wrapper algorithm "wraps" around learning algorithms and selects features based on learning performances, instead of examining correlations among data themselves. For a typical dataset, a wrapper algorithm performs a heuristic search through feature combination space. Searching can be carried out in two ways: forward selection and backward elimination. Forward selection starts with one single feature and

adds one more feature at each round (Figure 1.8), while back elimination begins with the full set of features and removes one feature at a time. A heuristic search is typically used to select the feature set that generates the best learning performance at each round to go to the next. The search is terminated when the learning performance stops improving (Figure 1.9).

Due to the involvement of learning algorithms in its evaluation schema, features selected by the wrapper algorithm perform well in many cases. However, the combination of the learning process with feature selection is also a potential shortcoming: biases of a specific learning algorithm can become embedded in feature selection. The resulting model may thus overfit training data and yield a disappointing performance with new instances. For the purpose of feature selection, however, it may not be a severe problem because it is the relative accuracies that determine the selection at each round, so the selected features are likely still the true informative features, despite the possibility of overestimating performance (Kohavi and John, 1997).

5. Evaluation of learning

The purpose of learning is to formulate a concise description that can be applied to all training data, and that correctly predicts the classification of an unknown instance. Naturally, the accuracy of classification will be the criterion to evaluate learning. Ideally, training data consists of a large number of instances that are statistically representative of instance distribution in the real world. The learning performance with ideal training data is likely a good indicator of prediction accuracy with future instances.

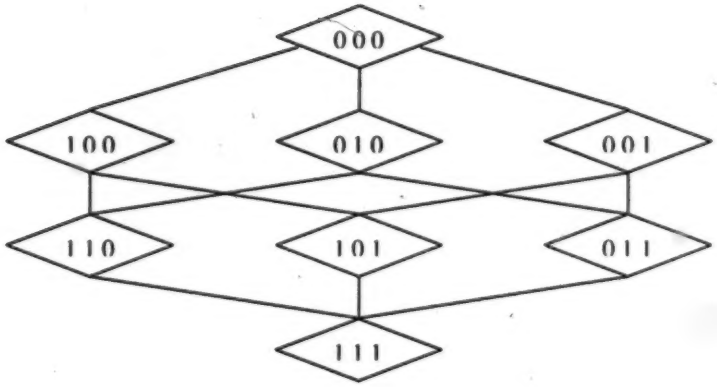


Figure 1.8: Forward selection in a wrapper. At each round, one more feature is added until all features are selected. We used this approach in the heuristic search.

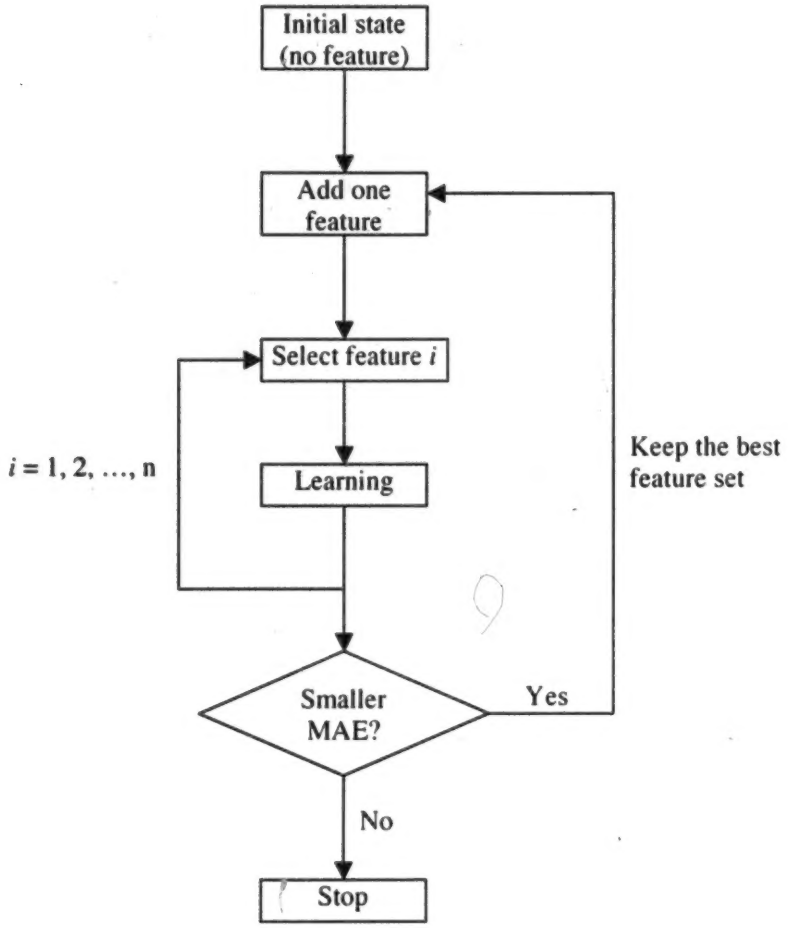


Figure 1.9: Heuristic search using a wrapper algorithm. Due to the computational complexity, not all feature sets can be retained at each round in the heuristic search in wrapper. In our study, we kept the top 200 feature sets at each round. The search terminates when the learning performance, measured by MAE (mean absolute error) using a repeated ten-fold cross validation, stops improving.

In reality, however, the statistical distribution of the training data is usually not known. In many cases, learning performance with training data is excellent but classification does poorly with new instances, a consequence called overfitting. Therefore, in many studies, a portion of the training data is held out so it does not participate in learning. They are only used to evaluate the model learned on the rest of the training data. Such a "hold out" process will generally provide a better estimate of the classifier's accuracy. This is being done for datasets with a reasonable number of instances.

For a small dataset that has a limited number of instances, however, there are too few instances to reserve a holdout set. Cross validation was thus developed to handle learning with a small dataset while still providing a reasonably objective estimate for future learning with new instances. In cross validation, the training data are divided into, for example, ten groups. The learning is carried out with nine groups of data while holding out the tenth group for evaluating the resulting classifier. This process is repeated ten times with each of the ten different groups as the holdout test set. The average of all ten learning performances will be considered as the estimated accuracy of learning with new instances. This is called ten-fold cross validation. Cross validation can maximize the number of data for training and accomplish a relatively objective evaluation for the generalized model.

In practice, different folds of cross validation have been used and results have varied depending on the specific datasets. Generally, ten-fold is considered a reasonable number to start with, although there is no theoretical basis for this rule. Moreover, if computing time is not a concern, a repeated ten-fold cross validation has been applied in

many learning tasks (Witten and Frank, 2000). Since a single cross validation can be influenced by the way the training data are divided, a repeated cross validation with different data partitions will likely remove this bias and generate an even better estimate of learning performance.

III. Thesis outline

In the present study, I collected structural and functional data of voltage-gated potassium channels (Chapter 2) (Li and Gallin, 2004), and attempted to deduce the structure-functional relationship of this group of ion channels. Using machine learning and feature selection techniques, I was able to identify a number of residues that are likely responsible for modulating the voltage sensitivity of voltage-gated potassium channels (Chapter 3). This computational model was then validated by permutation tests and independent test sets (Chapter 4). Based on these results, I made several VKC mutants and the functional characterization of these mutants using electrophysiology is in progress to further explorer the roles of these residues.

The current chapter is an introduction to the story of VKCs and the background of machine learning. Chapter 2 focuses on the collection of biological data on VKCs and the construction of a web accessible database, VKCDB. Chapter 3 describes the computational analysis of the sequence and functional data of VKCs using machine learning and feature selection. Chapter 4 details the validation of the computational model using both permutation tests and experimental data. Finally, I discuss the significance of the present study and potential future work in Chapter 5.

Chapter 2: Voltage-gated Potassium Channel Database

I. Introduction

Many biologists focus their research on one or a few specific protein families. While comprehensive sequence databases, such as GENBANK (Benson et al., 2004), and generally annotated databases, such as SWISSPROT (Boeckmann et al., 2003), are freely available, intensive studies on protein families are better supported by relatively small, focused databases that are developed for specific research needs. Our experience with the voltage-gated potassium channel database (VKCDB) provides an example of such a small, targeted protein family database. The approaches that we used to create this database are generally applicable to building databases for functional studies of other protein families.

Voltage-gated potassium channels (VKCs) are intrinsic membrane proteins that respond to changes in the transmembrane electrical field by altering conformation and selectively allowing potassium ions to pass through the membrane (Yellen, 2002). This property is the basis for VKCs' roles in shaping action potentials in neurons and modulating the electrical activity of excitable membranes. Mutations in VKC genes can lead to severe diseases, such as long QT syndrome and epilepsy (Towbin and Vatta, 2001; Kaneko et al., 2002). Thus, VKCs have been considered as possible targets for drug design (Cooper, 2001).

VKCs constitute a structurally and functionally diverse protein family. At this writing, there are over two hundred described members of this family from more than 35

A version of this chapter has been published. Li and Gallin 2004. BMC Bioinformatics 5:3.

organisms. VKC-related structural and functional data, particularly electrophysiological and pharmacological parameters, are distributed in dozens of databases and hundreds of journal articles. No single database contains structural and functional data for the various members of this large protein family. The application of comparative methods to the study of VKCs and other protein families depends on ready availability of both structural and functional data in an easily accessed database.

Here we report a customized database of VKC-related data that was created using semi-automated collection and management. This relational database currently holds 346 VKC entries. Each entry contains sequences, motifs, references, hyperlinks to other databases, and other available structural information. We have also collected available electrophysiological and pharmacological parameters for VKCs from several hundred published articles. These types of data are not properties of most proteins, and are not contained in general protein databases.

II. Construction and content

VKCDB was initially populated by performing a redundant set of searches of GENBANK for family members (Figure 2.1). GENBANK was first searched for protein sequences similar to the human Kv1.2 protein sequence (Ramaswami et al., 1990) using BLASTP (Altschul et al., 1997). The top 200 hits were used to perform BLASTP searches against GENBANK and SWISSPROT, yielding a comprehensive collection of VKCs. After collapsing all redundant BLASTP results, the top 319 non-redundant hits were collected; sequences with lower scores were not VKCs.

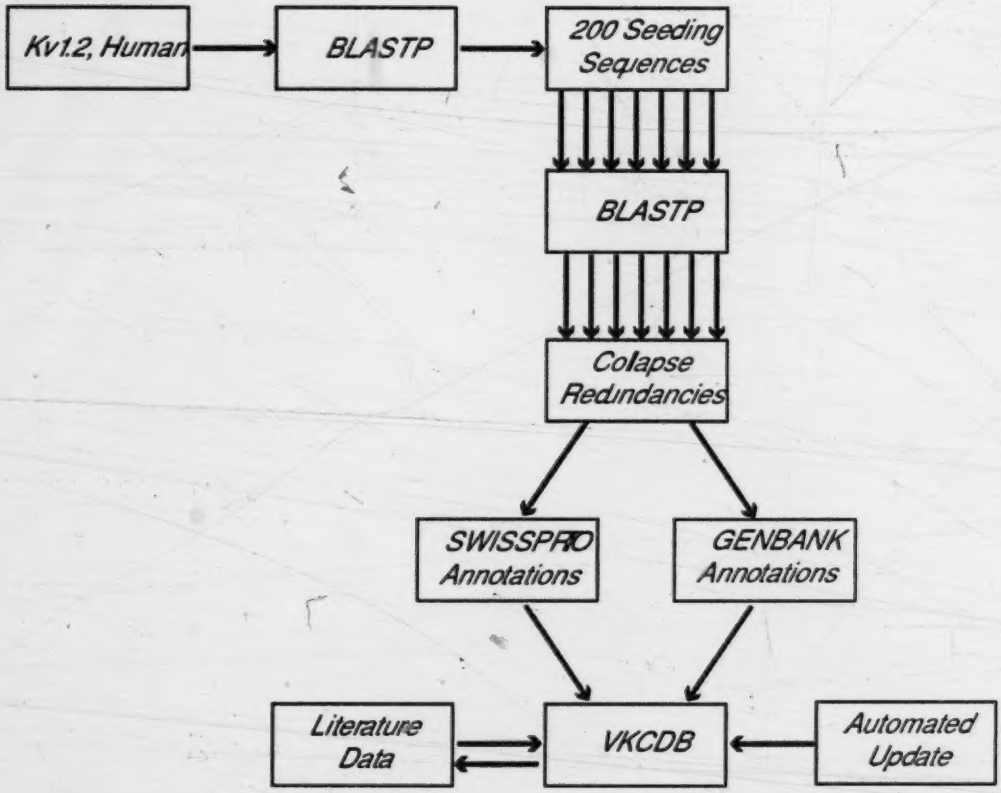


Figure 2.1: Populating VKCDB. VKCDB was populated by searching against GENBANK and SWISSPROT databases with 200 seeding VKC protein sequences. The redundancies in these results were collapsed, then structural and functional information was extracted from different databases and published articles using a combination of automated scripts and manual selection.

A Perl script was used to retrieve information on the 319 VKCs from GENBANK and SWISSPROT and store it in a MySQL relational database. A schematic diagram of the ER model of VKCDB can be found at VKCDB website (http://vkcdb.biology.ualberta.ca/images/ermodel.gif) (Figure 2.2). Data from redundant records in GENBANK and SWISSPROT were combined into a single entry. VKCDB entries with very similar sequences were manually checked, and their sequences were compared and annotated as "possible isoforms" or "sequence conflicts". Records for splicing variants were cross-referenced. Conflicting sequences that were submitted by different authors were cross-indexed as sequence conflicts, unless sequence errors were indicated in the literature, in which case the most recently updated sequence was kept. Entries labelled as "unknown products" from large sequencing projects that had the characteristic sequence pattern of the voltage sensor (a lysine or arginine residue at every third position of the fourth transmembrane domain) were used as BLASTP queries and annotated as members of a specific family of VKC based on the annotation of most similar BLAST results.

Using literature citations from GENBANK and SWISSPROT, we manually collected available electrophysiological and pharmacological data from published articles for each VKC entry. Conflicting data were all kept and hyperlinked to the references in PUBMED (McEntyre and Lipman, 2001).

All sequences were submitted to the TMHMM (Krogh et al., 2001) and PHD (Rost and Sander, 1994) servers for secondary structure prediction. Results from both analyses were parsed and combined into a single annotated sequence figure. This

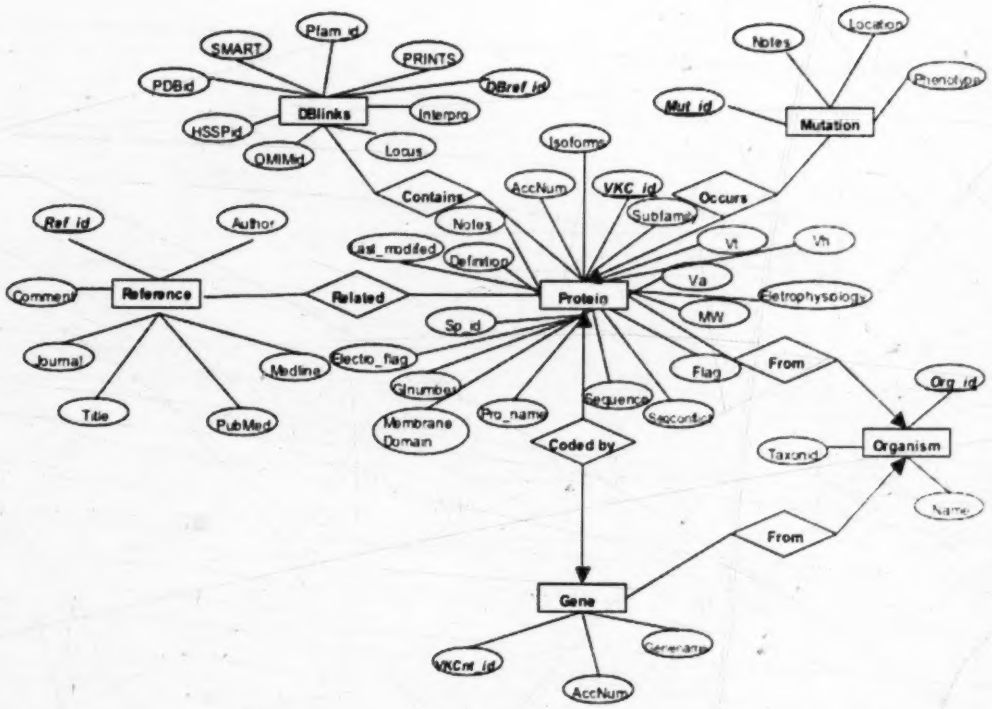


Figure 2.2: The ER model of VKCDB.

information is currently stored as a graphic because both of the prediction programs are not sufficiently accurate to be taken as a definitive result.

Sequences belonging to each of the four Kv families and the KCNQ family (Chandy, 1991) were extracted and a multiple alignment for each family was generated with ClustalW (Thompson et al., 1994). Alignments of the highly conserved regions (the T1 domain and the six transmembrane domains) were manually adjusted and included in VKCDB. Subsets of the aligned sequences can be selected and exported in FASTA format.

VKCDB is updated regularly. The "last modified" date of each VKCDB entry is compared to the corresponding field in the cognate GENBANK and SWISSPROT entries. Information from any of the archival entries that has been changed since the last VKCDB update is then parsed and used to update VKCDB. New VKC entries are collected by performing a BLAST search of GENBANK and SWISSPROT with all entries in VKCDB. The hits are combined into a non-redundant list for each subfamily and the top twenty scores on that non-redundant list that are not already entered in VKCDB are manually checked to confirm that they are indeed VKCs before adding them to VKCDB. If all twenty hits are VKCs, the next twenty hits are also manually evaluated; this process is repeated until non-VKC entries are found. Current entries in VKCDB were updated with SWISSPROT Release 43.1 (April 2004) and GENBANK entries as of April 2004.

III. Utility and discussion

At present VKCDB contains 346 VKC entries from 35 organisms. 39 VKCs are annotated as having between two and nine different isoforms, although some of these might be due to cloning or annotation artefacts. VKCDB can be browsed and searched through a web interface by several criteria, including VKC Kv subfamilies (Chandy, 1991), organism names, GENBANK Protein ID, protein description, reference information, and electrophysiological parameters. The VKC entries in all search and browse results can be individually selected on the web page to produce a batch sequence file in FASTA format for use in other applications. This site also implements a local BLAST server for similarity searches against VKCDB entries.

Each VKC entry page contains information such as protein accession number, protein name, protein sequence, coding gene name and accession number, SWISSPROT function description, references and hyperlinks to other biological databases. On each entry page, a button labelled "Electrophysiology" opens a pop-up window containing electrophysiological parameters, pharmacological data, and related references hyperlinked to PUBMED (Figure 2.3). We will be adding data on synthetic VKC mutants to VKCDB in the future, including links to the cognate wild type protein and the electrophysiological and pharmacological data. There is also a link to transmembrane helix predictions by TMHMM (Krogh et al., 2001) and PHD (Rost and Sander, 1994) on each entry page.

Multiple alignments of conserved regions of the four Kv families and of the KCNQ family are available on the VKCDB web site (Chandy, 1991), on the tools page

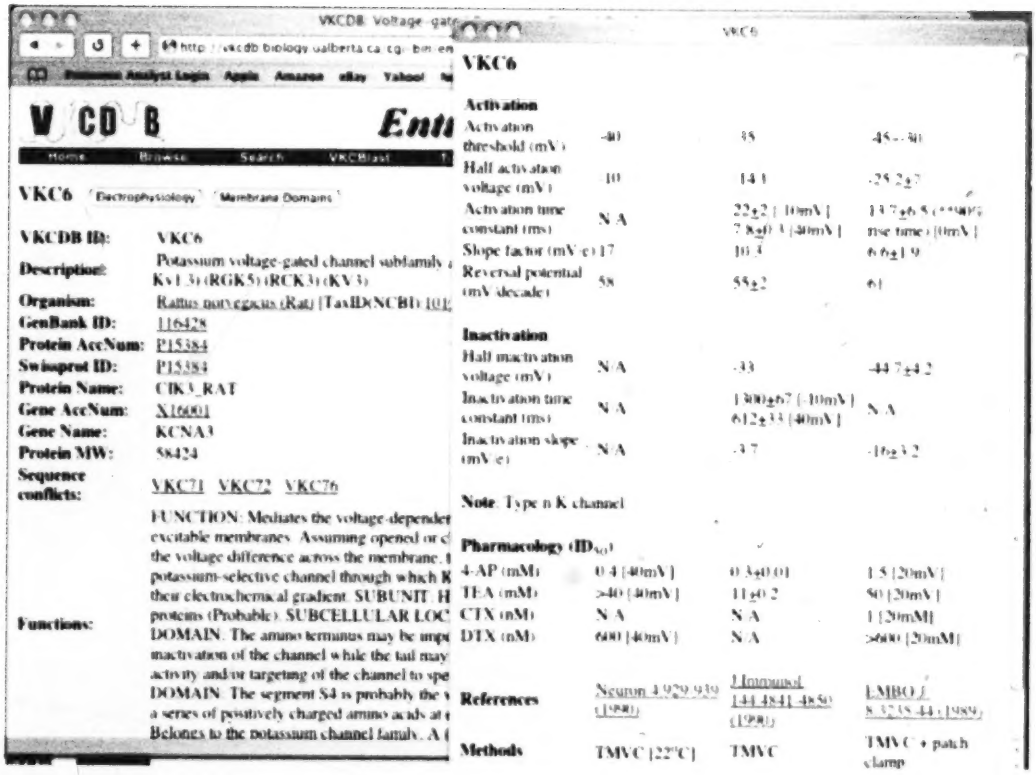

VKCDB Voltage-gat
V CD B
Home Browse Search VKCBlast
VKC6 Electrophysiology Membrane Domains
VKCDB ID: VKC6
Description:
Organism:
GenBank ID: 116428
Protein AccNum: P15384
Swissprot ID: P15384
Protein Name: CBK3_RAT
Gene AccNum: X16601
Gene Name: KCNA3
Protein MW: 58424
Sequence conflicts: VKC71 VKC72 VKC76
Functions:
FUNCTION: Mediates the voltage-dependent
DOMAIN. The amino terminus may be impo
DOMAIN. The segment S4 is probably the
Belongs to the potassium channel family. A
VKC6
VKC6
Activation
Activation threshold (mV) -40 -35 -45 - -30
Half activation voltage (mV) -10 -14.1 -25.2±2
Activation time constant (ms) N/A 22±2 [-10mV] 7.8±0.3 [-30mV] 13.7±6.5 (***30% rise time) [0mV]
Slope factor (mV/e) 17 10.3 6.6±1.9
Reversal potential (mV/decade) 58 55±2 61
Inactivation
Half inactivation voltage (mV) N/A -33 -44.7±4.2
Inactivation time constant (ms) N/A 1300±67 [-10mV] 612±33 [40mV] N/A
Inactivation slope (mV/e) N/A -4.7 -10±3.2
Note. Type n K channel
Pharmacology (ID50)
4-AP (mM) 0.4 [40mV] 0.3±0.01 1.5 [20mV]
TEA (mM) >40 [40mV] 11±0.2 80 [20mV]
CTX (nM) N/A N/A 1 [20mM]
DTX (nM) 600 [40mV] N/A >600 [20mM]
References Neuron 3:929-939 (1990) J Immunol 144:4841-4850 (1990) EMBO J 8:3235-44 (1989)
Methods TMVC [22°C] TMVC TMVC + patch clamp

Figure 2.3: Screen dump of the entry page of a VKCDB entry. The popup window contains the electrophysiological parameters of this entry. The transmembrane helix prediction by TMHMM and PHD can also be displayed for each VKCDB entry. Content of the entry page is extensively hyperlinked to various databases.

(http://vkcdb.biology.ualberta.ca/alignment.html). The individual sequences can be selected and downloaded, with gaps in place, in FASTA format for use in other applications.

The VKCDB web site includes a "submit" page that allows us to communicate with users on annotation errors, missing entries, and other information so that we can maintain accurate and updated VKC information in our database.

IV. Conclusions

VKCDB contains structural and functional data and related multiple alignments for voltage-gated potassium channels in a single database. The VKCDB web page is designed to provide easy access and searching through a user-friendly interface. It is also designed to interact easily with tools that we are developing to study the structure-functional relationship in VKCs using machine-learning approaches. The database information is also available as an XML file for users who wish to implement customized configurations.

Similar approaches can be taken to construct specific, small-to-medium-sized protein family databases, with minimum knowledge of Perl and MySQL database management. As a small, customized protein family database, VKCDB is a useful and convenient resource for research on VKCs. As our understanding of VKCs increases, more annotations and applications will be added to enrich VKCDB so that it can continue to serve as a main resource for structural and functional studies of VKCs.

V. Availability

VKCDB is freely accessible at http://vkcdb.biology.ualberta.ca. A snapshot of VKCDB in XML format can be freely downloaded from the website of VKCDB.

Chapter 3: Computational analysis of voltage-gated potassium channels

I. Introduction

During the evolution of proteins, there is interplay between selective forces acting to keep residue identities constant, thus preserving protein function, and selective forces that accept new variants of sequence that have altered properties conferring improved survival. Thus, when studying the evolution of structure-function relationships in a family of proteins, identification of invariant residues within the family identifies parts of the protein that are of central importance to its function. This idea is central to many comparative studies of protein structure/function relationships, and the concept has been extended to studies of pairs of residues whose identities co-vary in an apparently compensatory manner (Fleishman et al., 2004).

However, the converse idea, that varying residues are not centrally important to the protein's function, is not necessarily true. Although it is true that residues that do not have a major impact on protein function will show extensive variation over time, it is also true that residues that contribute to the quantitative variation in a protein's properties will also vary.

The problem that arises, then, is how to distinguish the residues whose variation is responsible for functional variations in the protein from those residues whose variation is relatively immaterial to function. These residues will not be detected by evaluating the amount of variation in a given residue or in pairs of residues. Rather, the residues will co-vary with the property of the protein that they affect. To solve this problem it is necessary to use techniques that can detect associations between residue identities at any

position in the protein and the quantitative value of the parameter of interest. This chapter describes an analysis to detect such structure-functional association in voltage-gated potassium channels (VKCs) using machine learning techniques.

VKCs are membrane proteins that regulate the passage of potassium ions through membranes (Yellen, 2002). When the voltage difference across a membrane reaches a threshold, the probability that VKCs will open begins to become significant, allowing potassium ions to diffuse through an ion-selective pore in the channel. This voltage-regulated potassium ion permeability is critical to cellular excitability. Mutations in VKC genes have been shown to be associated with cardiac arrhythmias (Jentsch, 2000), episodic ataxia (Comu et al., 1996), and other diseases (Abdul and Hoosein, 2002a; Koni et al., 2003).

A functional VKC consists of four subunits, each containing six transmembrane regions, S1 through S6. S4 has been shown to function as the main voltage-sensing domain (Yellen, 2002), acting by moving through the membrane upon depolarization (Larsson et al., 1996; Jiang et al., 2003b). Through an unknown mechanism, this movement causes a conformational change in the region of the pore, likely in S5 and S6, to open the "gate" and allow potassium ions to pass through.

A great deal has been learned of the molecular mechanisms of VKC function in the last ten years (Doyle et al., 1998; Bixby et al., 1999; Sokolova et al., 2001; Jiang et al., 2002a; Jiang et al., 2003a; Kuo et al., 2003). In the absence of accurate three-dimensional structures of various VKCs at different opening/closing stages, mutagenesis of individual residues of different VKCs has been the main method for inferring the structure-function relationship of VKCs. However, it is prohibitively time-consuming

and costly to do mutagenesis of all residues individually and in combinations in different VKCs. Therefore, computational tools, usually multiple sequence alignment, have been used to identify conserved regions of VKCs and limit the priority in mutagenesis experiments to evolutionarily conserved residues (MacKinnon, 1991b; Miller, 1991; Heginbotham et al., 1992). Unfortunately, details of the elaborate structure-function relationship between individual residues and the electrophysiological properties, which are mostly continuous quantitative parameters (Hille, 2001), are too complicated to understand by simple inspection of aligned VKC sequences. With dozens of VKC sequences of a few hundred residues each and continuous electrophysiological variables, more mature data mining tools, such as machine learning, are necessary.

Machine learning generalizes the underlying data model by "learning" from the existing data using various classification rules. It yields a mathematical model that can best describe the existing data and predict classifications of new data (Mitchell, 1997). Because of its ability to extract complex models from large datasets, machine learning has been successfully applied to many data-rich problems such as marketing reports, weather prediction, automatic genome annotation and microarray data analysis (Tag and Peak, 1996; Hayes and Borodovsky, 1998; Bose and Mahapatra, 2001; Ringner and Peterson, 2003).

Typically, a protein family comprises dozens of members with hundreds of residues in each member. Such datasets present a characteristic type of problem for machine learning. First, a typical training dataset for machine learning contains distinctively labeled "features" in every instance. With protein sequence datasets, all data have to be pre-processed to determine which residues of all sequences should be aligned

with each other to identify homologous residues (features). Second, dozens of sequences with hundreds of residues each create a dataset with very high dimensionality, which compromises learning performance. Finally, besides generating a classifier with high accuracy, it is pertinent to bench biologists to evaluate the biological importance of individual residues (features) that contribute to a good learning performance during training. Therefore, it is desirable to use learning methods that return the basis for their prediction.

I have mined the available VKC sequence and electrophysiological data using machine learning and related feature selection techniques, and derived a model that predicts one of the central electrophysiological parameters, half activation voltage (V_{50}) (Hille, 2001), of a given VKC, based on only its amino acid sequence. The best result was obtained using a k-nearest neighbor classifier (k = 1) combined with a wrapper algorithm for feature selection (Kohavi and John, 1997), yielding a mean absolute error (MAE) between the predicted and published V_{50} values of 7.0mV in a repeated ten-fold cross validation. The training process also provides a rational basis for identifying residues potentially critical to the activation of VKCs, and several identified key residues are located in regions that have been proposed to modulate VKC activation.

Recently, a complementary computational approach was applied to identify residue pairs of VKCs that co-vary during evolution, and some of the identified residues have been shown to be "gating-sensitive" (Fleishman et al., 2004). As expected, these evolutionarily conserved residues are mostly located in the functionally critical domains of VKCs, the S4-S6 (Fleishman et al., 2004). The present study detected those evolutionarily varying residues whose variations lead to functional diversity of VKCs. It

is thus not surprising that they all reside in the S1-S3 helices, which modulate the critical voltage sensing and gating function of VKCs but are not part of the ion pore or voltage sensor (Yellen, 1998). A similar approach can be used to generate biological hypotheses in other protein families and these hypotheses can be practically tested using site-directed mutagenesis.

II. Methods

1. Dataset

Data used in this project were drawn from VKCDB, a voltage-gated potassium channel database (Chapter 2) (Li and Gallin, 2004). 58 VKC sequences with associated half activation voltage (V_{50}) values were extracted from VKCDB; most of the sequences have more than 500 amino acid residues. All published V_{50} values used in this study were experimentally determined under similar experimental conditions, using a two-electrode voltage clamp in *Xenopus* oocytes (Hille, 2001). Averages were used for those VKCs for which different V_{50} values have been published by different groups (Stuhmer et al., 1989; Schroter et al., 1991; Rettig et al., 1992; Scholle et al., 2000).

All sequences were aligned with PepTool (Wishart et al., 1994), followed by manual adjustment. Because there is large sequence variation at both termini and some loop regions of the VKCs, only blocks of residues that contained relatively few gaps were kept for analysis (Dataset 1). These blocks are more conserved and likely contain residues that are functionally important to all aligned VKCs, assuming functionally critical residues are more conserved during evolution.

2. Problem formulation

To formulate the problem into a typical supervised learning task, the dataset was considered as a training set with 58 instances. Each of the alignment positions was taken as one nominal attribute (feature), and all attributes were assumed to be independent of each other. The order of residues (features) was not taken into account during learning. In numerical prediction analyses, the classes were the real V_{50} numerical values. In categorical prediction analyses, V_{50} values were divided into seven nominal classes based on their values; $-50 > V_{50} \geq -30$mV, $-30 > V_{50} \geq -20$mV, $-20 > V_{50} \geq -10$mV, $-10 > V_{50} \geq 0$mV, $0 > V_{50} \geq 10$mV, $10 > V_{50} \geq 20$mV and $20 > V_{50} \geq 65$mV. The goal is to extract the data model that can best describe the relationship between the (attributes) features and the labeled classes of these data, and correctly predict the class or the numerical value of V_{50} of any given VKC sequence (Figure 2.1).

3. Basic learning algorithms

The KNN (k-nearest neighbor) classifier was used in both numerical prediction and categorical prediction analysis. All KNN classifications were tested with k values of 1 to 5. The best performances were always obtained when k is 1. Decision Tree, Naïve Bayes classifier, kernel density classifier and OneR classifier were also used in categorical predictions. The algorithms used are implemented in the WEKA package 3.2.3 (Witten and Frank, 2000).

The prediction accuracies were used to evaluate the learning performance in categorical prediction. The mean absolute errors (MAEs), the average absolute difference between the predicted values and the published values, were used to assess the numerical prediction. All learning performances were evaluated using a repeated ten-fold

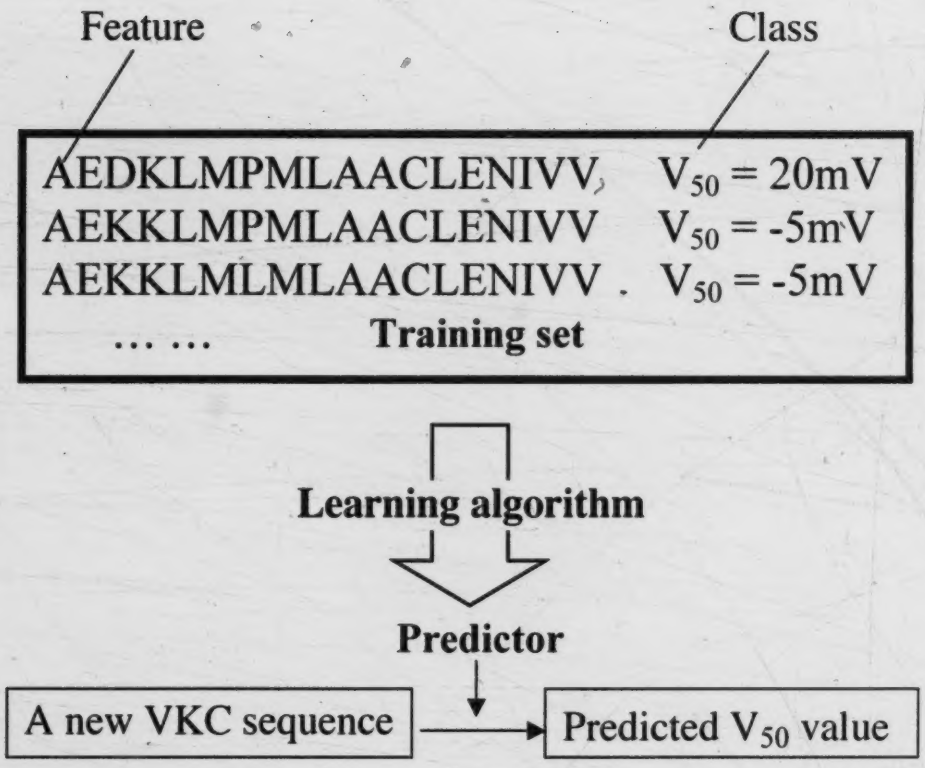


Figure 3.1: Problem formulation. Our training dataset contained 58 VKC sequences. Each aligned position was one feature. V_{50} values were the final class that was to be predicted. Learning algorithms were trained with these data and generated a predictor (model) that could predict the V_{50} value based on only the amino acid sequence of a VKC.

cross validation. In other words, the dataset was randomly split into ten partitions for a ten-fold cross validation, and the process was repeated for ten times and the average of the MAEs of these ten ten-fold cross validations was used to evaluate the learning.

4. Feature selection

Because high dimensional datasets with large number of irrelevant features can compromise learning performance, two feature selection methods, the filter and wrapper algorithms, were implemented.

Filter uses information gain as a criterion to rank features in a dataset, and only the top-ranked features are used in learning (Almuallim and Dietterich, 1991; Kira and Rendell, 1992; Cardie, 1993). Different numbers of top-ranked features were selected for learning, and the sets that produced the best learning performance were considered the best feature sets using the filter algorithm.

The wrapper approach to feature selection screens features in a dataset and selects the "relevant" features based on learning performances (Kohavi and John, 1997). Forward selection was used in this approach, in which one feature (residue) was added at each round (Figure 1.8), until learning performances stopped improving (Figure 1.9) (Kohavi and John, 1997). To avoid having the performance trapped in a local optimum, the top 200 feature sets based on their learning performance were kept at the end of each round and were used as starting points for the next round of selection. Despite the existence of redundant feature sets, the number of non-redundant feature sets was well above 100 at each round. The search was continued for five rounds after the learning performance stopped improving to ensure that performance had plateaued.

5. Residue swapping

I also applied a "residue swap" heuristic, similar to the branch-swapping step used to construct phylogenetic trees (Adams, 1972), to try to further improve the prediction accuracy. For the best feature set selected by the wrapper, each residue was sequentially replaced with every other residue that was not in the final set, and the new feature combination was evaluated for prediction accuracy using a repeated ten-fold cross validation.

6. Distance matrices in k-nearest neighbor classification (KNN)

A KNN classifier is a set of n-dimensional vectors (where n = the number of features) to which new instances are compared (Mitchell, 1997). It classifies a new instance by evaluating its distance from each of the classifier instances and chooses the class label of the classifier instance that is closest to the new instance as the predicted class of the new instance (Figure 1.7). For more than one classifier instance with an identical distance to the new instance, one of the class labels of these classifier instances is randomly picked and assigned in categorical predictions; averages of equidistant classifier instances are calculated for numerical prediction.

The distance between any two vectors is obtained by taking the sum of the square of the distances between all pairs of attributes (dimensions). For nominal attributes, such as amino acid residues, the KNN algorithm can simply take 1 and 0 as the distance between a pair of different and same residues, respectively. I also implemented the KNN algorithm to incorporate PAM (Schwartz and Dayhoff, 1978) and BLOSUM (Henikoff and Henikoff, 1992) matrices as a measure of distance between pairs of features (residues) of two VKC sequences (Formulas 3.1 and 3.2). Since the scores in amino acid comparison matrices go up when two amino acid residues are more similar to each other,

which is the opposite to distance measurement in KNN classification, I converted amino acid comparison scores accordingly (Formula 3.0).

To convert scores in the BLOSUM62 or PAM100 matrix:

score_range = highest score – lowest score

converted $score_i$ = score_range - (original_$score_i$ – lowest score) (3.0)

$$f1: D = \sum_{i=1}^{n} score_i^2$$

D: Distance between two instances.

n: Number of features.

$$\text{Identity matrix: } score_i = \begin{cases} 1 & \text{if features of two instances are different} \\ 0 & \text{if features of two instances are the same} \end{cases} \quad (3.1)$$

Other matrices: $score_i$ = converted $score_i$ from pairwise comparison (3.2)

7. Outlier selection

Va values for the various VKCs have been measured and published by different labs. The techniques that were used to obtain V_{50} are not fully standardized. To minimize the effect of possible outliers, another best-first search was performed. One VKC sequence was deleted from the training set at each round, and the learning was carried out with the remaining VKC sequences. The deleted sequence was considered an outlier if the remaining dataset yielded better learning performance than the full dataset. The search stopped if the learning performance no longer improved after a further round of deletion. Due to computational complexity, the outlier selection was not combined

with full feature selection of the wrapper algorithm (Kohavi and John, 1997). Instead, the best feature set selected by the wrapper algorithm was applied to outlier selection.

8. Final predictor construction

The training dataset contained 58 VKC sequences. Based on the effect of individual VKC sequences on the overall learning performance, four sequences were identified as possible outliers; the remaining data formed a new dataset (Dataset 2) with 54 VKC sequences (Figure 3.2). During the training process using Dataset 2, one best feature (residue) set was selected by the wrapper algorithm to predict the V_{50} values with an MAE of 7.0mV using a KNN classifier (k = 1). One predictor was then constructed, using Dataset 2, the best feature set, the BLOSUM 62 scoring matrix and the KNN classification (k = 1).

To predict the V_{50} value of a new query sequence, the query sequence is first aligned with the profile alignment of Dataset 2 using ClustalW (Higgins et al., 1996). The residues at the aligned selected positions are extracted to produce a data file for V_{50} prediction.

III. Results

1. Learning without feature selection

A dataset containing 296 aligned positions from 58 VKC sequences (Dataset 1) was used to train different learning algorithms to predict the V_{50} value of a given VKC sequence (Figure 3.3A). V_{50} values were divided into seven nominal classes. The best categorical learning performance was below 30% accuracy (Figure 3.3A). The MAE of the best numerical prediction of V_{50} values with the KNN classifier (Figure 3.3B) was close to 18mV. Evidently, these learning algorithms alone do not produce an acceptable

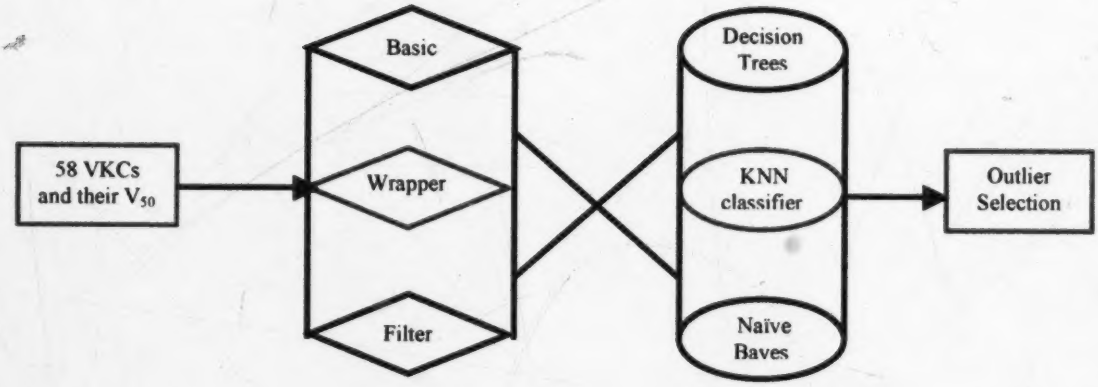


Figure 3.2: Flow chart of procedures followed to evaluate factors contributing to optimal V_{50} prediction. Different feature selection methods were tested with different learning algorithms. The best learning performance was obtained using a KNN classifier with a wrapper algorithm for feature selection, combined with outlier selection.

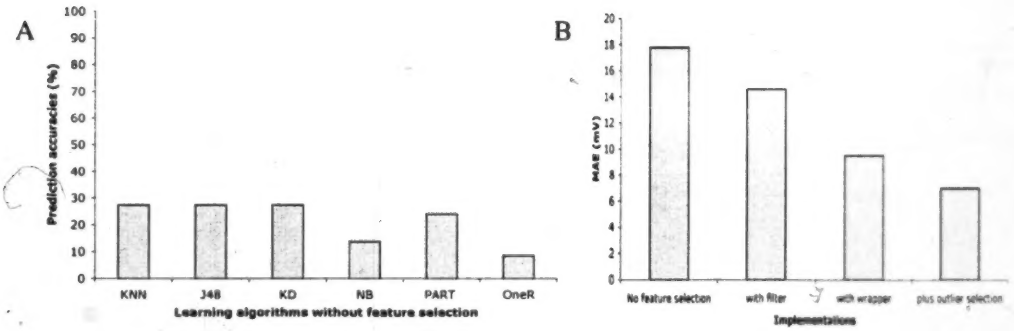


Figure 3.3: Learning performances with different algorithms and different implementations. All bars represent results of a repeated (ten time) ten-fold cross validation.

A: Categorical learning with different learning algorithms without feature selection. The V_{50} values were divided into seven classes based on their values; the learning was done without feature selection.

B: Improvement of KNN prediction accuracies in different implementations. Results of KNN classification without feature selection, with the filter algorithm, with the wrapper algorithm, and with outlier selection are shown. Both feature selection algorithms improved learning performance. The best learning accuracy was obtained using the KNN classifier combined with wrapper. It yields a mean absolute errors of 7.0mV with the new dataset (Dataset 2) of 54 VKC sequences after four possible outliers were deleted from the original dataset (Dataset 1).

model for prediction if they are trained with such a high dimensional dataset of less than 60 instances.

2. Learning with a filter algorithm

To improve learning performance with this high dimensional dataset, I added feature selection, using a filter algorithm, before learning. All residues (features) were ranked based on their information gain scores (Almuallim and Dietterich, 1991; Kira and Rendell, 1992). Based on the ranking, different numbers of residues were used for learning. The best learning performance was obtained using only five features (residues) with top ranking, and the accuracy improved to 36%. The MAE of the numerical prediction of V_{50} values with a KNN classifier was now at 15mV (Figure 3.3B). While dimension reduction by the filter algorithm did appear to yield a better learning performance, the prediction accuracy was still not satisfactory.

3. Learning with a wrapper algorithm

I also applied the wrapper algorithm, a more learning performance-driven feature selection method than the filter algorithm (Kohavi and John, 1997). From a large number of sets of residue (features) combinations, a wrapper algorithm selected the residue set that yielded the best learning performance. The prediction accuracies with all categorical learning algorithms improved, with the best classification of 60% accuracy using the KNN classifier. When the KNN classifier (k = 1) was combined with the wrapper algorithm to predict a numerical V_{50} value based on a VKC sequence, the MAE of prediction improved to 9.5mV from 17.8mV (Figure 3.3B). The best prediction accuracy was obtained with six residues (features).

I used a transformed BLOSUM62 amino acid matrix for distance measurement in KNN classification (Formula 3.1 and 3.2). I also tried the PAM100 matrix and a simple identity matrix (Formula 3.1 and 3.2). The best MAEs remained unchanged in a repeated ten-fold cross validation. Compared with results using the BLOSUM62 matrix, different but overlapped sets of features (residues) were selected using the PAM100 and identity matrices (Table 3.1).

4. Learning combined with outlier selection

Since the dataset has only 58 VKC sequences, a small number of outliers or incorrect class labels might have greatly affected the training process and thus led to poor learning performance. I evaluated the effect of deleting each sequence from the dataset, by training the KNN classifier with each of the 58 possible sets of 57 sequences. The top 50 subsets with 57 VKC sequences that produced the best learning performances using a repeated ten-fold cross validation were kept and the pruning procedure was then repeated with each of the 50 subsets as a starting point (Figure 3.4A). The six feature set that gives the best learning performances using Dataset 1 (MAE = 9.5mV) was used during outlier selection. Despite the plateau in Round 1 and 3, there were significant improvement of learning accuracies in Round 2 and Round 4. After four pruning rounds the improvement in accuracy significantly slowed down in the following rounds (Figure 3.4B). Thus, we believe that Round 1-4 represents informative gains in accuracy from deleting true outliers, whereas the improvement in later rounds was likely due to overfitting.

During the pruning process, four VKC sequences, VKC8 (Kv1.3 mouse), VKC98 (Kv1.4 dog), VKC149 (Kv2 squid), and VKC171 (Kv4.3 mouse) (Chapter 2) (Li and Gallin, 2004), were consistently selected as "outliers" from Round1-4, although the order

Distance matrices	Selected residue sets
BLOSUM62	97, **100**, 117, 125, 135, **154**
PAM100	83, 95, 97, **100**, 117, 131, 141, **154**
Identity matrix	83, 92, 95, **100**, 103, 123, 135, **154**, 273

Table 3.1: "Best" feature (residue) sets selected by the wrapper algorithm with different distance matrices in a KNN classifier. Residues that were selected with more than one matrix were underlined, and residue **100** and **154** were selected with all three distance matrices.

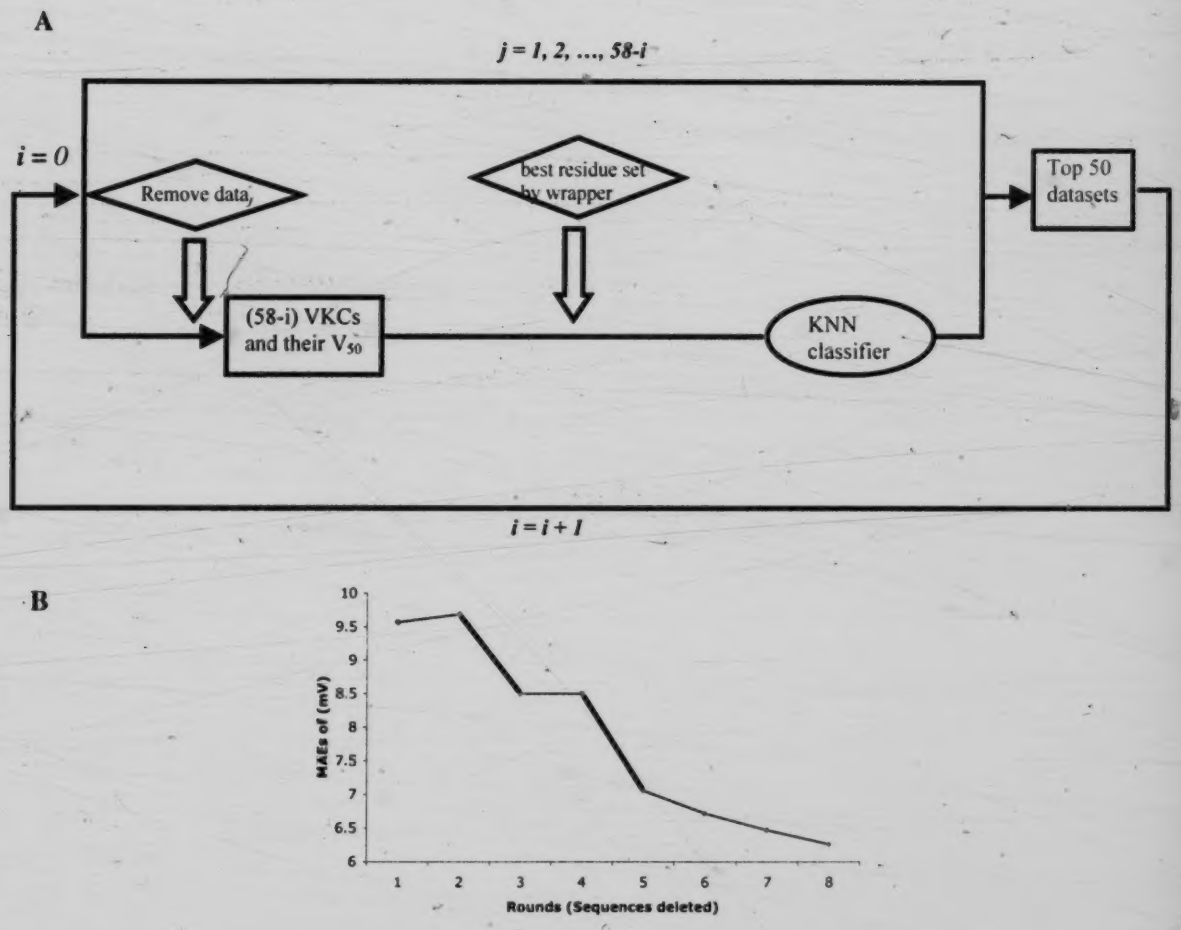


Figure 3.4: Schematic of the process of outlier selection, and the variations in MAEs during outlier selection using KNN classifier.

A: Each instance of Dataset 1 was individually deleted to select the resulting datasets that produce improved learning performances. The top 50 new subsets were kept at each round, and individual deletions were repeated. The best feature set selected by the wrapper algorithm as described in the paper was used in training.

B: Variation of learning performance using KNN classifier during outlier selection. The mean absolute errors of prediction improved with selective removal of putative outlier instances. There was a significant improvement of learning accuracies at Round 2 and 4 (highlighted in solid dark). After Round 4, the improvement of learning performances slowed down significantly.

by which they were "deleted" varied. I therefore deleted them to create a new dataset of 54 sequences (Dataset 2). The new dataset was used to construct the KNN final classifier, for which the best MAE improved to 7.0mV (Figure 3.3B). I also re-ran the wrapper algorithm with Dataset 2, and exactly the same feature set was again selected, yielding the best MAE of 7.0mV.

5. Identification of functionally critical residues

The wrapper algorithm identifies a relatively small number of residues that are the primary determinants of accurate learning. With both Dataset 1 (58 instances) and Dataset 2 (54 instances), six residues were consistently selected to produce the best learning performances (Table 3.1), using a KNN classifier and BLOSUM62 matrix. I reason that the residues that were identified as most informative in learning are more likely involved in the physical activation process of VKCs. Selected residues were mapped onto a schematic of the S1-S6 structure (Figure 3.5). All of them reside in S1-S3, a region that likely plays a modulating role in VKC functioning (Yellen, 1998; Treptow et al., 2004).

IV. Discussion

1. Learning with high dimensional data

Data with high dimensionality are a "curse" to learning performance. As a rule of thumb, the number of instances should be no less, and preferably more, than the number of features to obtain a reasonable learning accuracy (Kohavi and John, 1997). Even with a large number of instances, a large number of irrelevant features can still compromise the learning performance (Kohavi and John, 1997). For biological data, however, enough

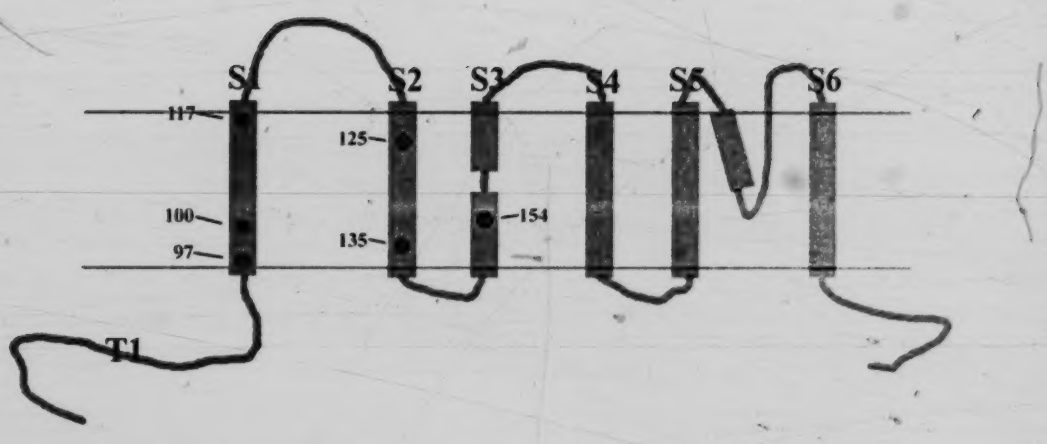


Figure 3.5: Informative residues identified by wrapper. The six residues selected by the wrapper algorithm yielded the best learning performance (MAE = 7.0mV). All six of them are located in S1-S3 helices, a region that is considered to play secondary roles in voltage sensing.

examples with relatively small dimension are not always achievable. Without dimension reduction or feature selection prior to data analysis, learning performances with high dimensional data are poor. Many dimension reduction methods have thus been applied to improve learning performance, including principle component analysis and linear discriminant analysis (Mendez et al., 2002; Nguyen and Rocke, 2002).

I faced this problem in my analyses. There are fewer than 60 VKC sequences with published V_{50} values, and there are nearly 300 residues in each sequence alignment after trimming poorly conserved regions. Most residues likely have little or no role in determining V_{50} values, and thus are "irrelevant features". Training without feature selection using several machine learning algorithms yielded prediction accuracies consistently lower than 30% (Figure 3.3A).

Application of the filter algorithm before learning improved the accuracy marginally (Figure 3.3B). The filter approach is a pre-learning data processing method; it is based on pre-learning evaluation of the information content of the dataset, and thus is independent of the training process. It has been successfully used in other tasks to obtain better learning performance (Almuallim and Dietterich, 1991; Kira and Rendell, 1992). However, it may or may not select all the true relevant features depending on the datasets and the selection criteria. Considering the number of features and number of instances in the datasets, some irrelevant features may well correlate with the final class labels by chance and display a high information gain potential, which will not be distinguished by filter. It is thus not very surprising that the filter algorithm did not perform well with this dataset.

I then applied a wrapper algorithm to select features during the learning process. The wrapper algorithm uses a heuristic search to select the feature combinations that yield the best learning performance (Kohavi and John, 1997). It "wraps" around the learner and selects the best feature sets based on learning accuracies. To avoid being trapped in local optima during heuristic search, I selected the top 200 residue combinations at each round, used them all as starting points for the next round of searching. Best-first searching was continued until learning performance stabilized. The learning accuracies with the wrapper algorithm increased greatly for all learning algorithms I used. The best categorical result was obtained with the KNN classifier (k = 1) with an accuracy of 60%. To generate a predictor that can predict a numeric V_{50} value from a query sequence, I also trained the KNN classifier combined with a wrapper algorithm for numerical classification; the mean absolute error of prediction improved to 9.5mV from 17.8mV (Figure 3.3B).

Since the wrapper algorithm does not use an exhaustive search and does not guarantee optimal feature selection, I applied "residue swapping" to identify residues that yielded better results in the context of the finally selected residue set. However, residue swapping did not produce any new feature sets that yielded better predictive performance. On one hand, as an empirical operation, branch swapping does not guarantee the global optimum in phylogenetic analysis (Adams, 1972). On the other hand, although I could not use an exhaustive search in wrapper, the 200 top feature combinations were kept at each round of best-first search, to increase the possibility of achieving a close-to-optimal solution even without residue swapping. The best feature (residue) set remained unchanged after residue swapping, which provided another piece

of evidence supporting the idea that residues selected by the wrapper algorithm are likely true "relevant" features that establish the final classification.

Thus, using the wrapper algorithm greatly improved the learning accuracies with the dataset. However, since the wrapper algorithm selectively searches the feature space based on the learning performance from the previous round, I can not entirely exclude the possibility of some degree of overfitting, in spite of the evaluation by a repeated ten-fold cross validation.

2. Outlier selection

Typically, with a sufficient amount of data, classification using machine learning is expected to be relatively insensitive to outliers. However, a low number of instances relative to features of structural and functional data can increase susceptibility to outlier effects.

The V_{50} values in the dataset were obtained from publications from dozens of labs. I used averages for three V_{50} values in the datasets because different investigators have published different V_{50} values for the same VKC sequences. The difference in V_{50} values of the same VKCs from different labs sometimes exceeds 15mV or more (Stuhmer et al., 1989; Schroter et al., 1991; Rettig et al., 1992; Scholle et al., 2000). Thus, it is almost certain that some VKCs in the datasets compromise learning because they are incorrectly labeled.

I evaluated the prediction accuracies with datasets from which one sequence was pruned at each round of training (Figure 3.4A). Based on the variations in learning performances, I stopped at Round 4 (Figure 3.4B). At both Round 2 and 4, learning performances displayed an improvement of MAE of almost 1.5mV (Figure 3.4B). The

improvement of learning performances after Round 4 decreased significantly (Figure 3.4B).

One sequence was deleted in each round of pruning. Creating the best learning performance from the remaining data, the top fifty such "remaining" sequence sets were all used as starting points for the next round of searching. Four sequences were consistently selected for "deletion" in the first four rounds, although sometimes in different order. The best learning performance produced a MAE of 7.0 mV with the new dataset of 54 VKC sequences (Figure 3.3B), after deleting four potential outliers.

Outliers may arise from experimental errors or the channels may be activated by different mechanisms so V_{50} values would be affected by different set of residues from the non-outliers. In the latter case, the "deleted" outliers become interesting research targets (Overturf et al., 1994; Holmqvist et al., 2002a). However, I could not rigorously exclude the possibility that they were selected as outliers due to the specific dataset I used and possible data overfitting in the training. More experimental work needs to be done to clarify these issues.

Among the four deleted "outliers", two different values of V_{50} were reported in the paper that characterized VKC8 (Kv1.3 mouse) (Grissmer et al., 1990), which could be due to errors from experiments or other sources. VKC149 is a squid Kv2 channel. Its G-V curve, which was used to obtain its V_{50} value, had to oddly fit with two Boltzmann functions, adding another layer of complexity to its gating mechanism (Patton et al., 1997). Both VKC171 (Kv4.3 mouse) (Holmqvist et al., 2002b) and VKC98 (Kv1.4 dog) (Vogalis et al., 1995) are fast inactivating Kv4 channels. Their activation might overlap

their inactivation, which makes it difficult to obtain an accurate V_{50} value (Holmqvist et al., 2002b).

3. Identification of biologically important residues (features)

While I have focused on building a model that can predict the V_{50} value of a given VKC sequence with reasonable accuracy, I am equally interested in identifying the residues that are involved in modulating voltage sensitivity of VKCs. Statistical modeling has long been used to solve problems in different fields. It is still debatable, however, if optimized data models reflect underlying mechanisms and if features used in data models are indeed the contributing factors in reality (Breiman, 2001). Evidently, where the truth lies depends on the nature of the problem and the datasets at hand.

In the training using a KNN classifier with a BLOSUM62 matrix, different feature (residue) sets were selected by the wrapper algorithm and screened to identify the feature set that yielded the best learning performance (Figure 3.5). In a forward selection approach, one feature (residue) was added at each round. Although different features were sequentially selected in different orders during the first five rounds, the feature set that produced the best learning performance converged to six residues. These six residues are the best features in predicting the V_{50} value of VKCs based on their amino acid sequences. Likely, these residues are central to functional determination of the voltage sensitivity of VKCs.

Despite similar learning performances, different but overlapped residue sets were selected when using different distance matrices including PAM100 (Table 3.1). It is possible that, due to the non-exhaustive nature of heuristic search, feature selection results by the wrapper algorithm were not the optimal ones, and thus not all the true

"best" features were identified. However, residues that were insensitive to distance matrices and consistently selected are evidently more likely to be biologically important (Table 3.1). I will focus my discussion on the six residues selected when using BLOSUM62 matrix in KNN classification.

Some functionally important residues may not be identified using this approach. If a group of residues co-vary because they interact with each other to affect V_{50} values of VKCs, for example, after one residue is identified, the addition of the other residues may not further improve learning performances, and thus they would not be selected. However, no feature co-varied precisely with the six selected features in a covariation study (Gallin, unpublished result). Also, the datasets contain a tiny subset of all the VKCs in nature, which may not be an unbiased representation of all VKCs, so the residues that are selected may be only pertinent to the specific datasets. The quality of the experimental data are also a factor, indicated by the different V_{50} values obtained by different research labs for the same VKC (Stuhmer et al., 1989; Schroter et al., 1991; Rettig et al., 1992; Scholle et al., 2000). Outlier selection may have helped alleviate the problem, but it is still a potential error source. Nevertheless, the combination of the selected residues should be a good indication of potentially functionally important structure elements.

In a recent study, evolutionarily conserved residue pairs, which are presumably involved in the essential functions of VKCs, were computationally identified and most of these residues are located in the so-called core functional elements (S4-S6) (Fleishman et al., 2004), the pore region and the voltage sensor. In contrast, the approach to structure/function analysis in this thesis is aimed at identifying structural elements that modulate the voltage sensitivity, not those that are essential for voltage sensitivity. While

S4 is considered the main voltage-sensing unit, S1-S3 is thought to play a modulating role in the voltage sensitivity of VKCs (Yellen, 1998; Treptow et al., 2004). Consistent with their functionally modulating roles, all residues selected in this study are indeed located in S1-S3 region (Figure 3.5). It is not surprising that several other residues in the S1-S3 region that were not selected in the present study were also shown to modulate the voltage sensitivity of VKCs (Gallin, unpublished result and (Papazian et al., 1995; Tiwari-Woodruff et al., 1997)). Most of these residues were demonstrated to interact directly with the positively charged key residues in the voltage sensor (S4 helix) and closely involved in the movement of the voltage sensor. They are thus a part of the core structural element, which is indirectly supported by the fact that they are highly conserved among VKCs. Consequently, there is not enough variation in the dataset at these positions to associate them with the functional variation.

More discussion on the possible functional roles of these identified residues in VKC activation can be found in Chapter 5.

V. Conclusions

Machine learning methods have been widely used in biological analyses because of their capacity for dealing with data-rich tasks. Using a dataset of 58 VKC sequences with their V_{50} values, I built a predictor of the V_{50} value of a given VKC based on its amino acid sequence. Despite the limited number of training data and uncertain quality of physiological data, an MAE of prediction of 7.0mV was obtained using a KNN classifier combined with a wrapper algorithm for feature selection (Figure 3.3B). The prediction was evaluated by a repeated (ten times) ten-fold cross validation. As more data become available from ongoing isolation and characterization of VKCs, better prediction

is expected. During training, four possible outliers were singled out and removed from the training set to improve the learning performance (Figure 3.4). Several residues with potential biological implications were identified for further study (Figure 3.5).

The analysis presented in this chapter demonstrated that machine learning methods can be productively applied to structure-functional study with datasets of limited size. The preliminary analyses of this type of question will provide biologists useful tools to predict certain biological functions with a reasonable accuracy. With knowledge of the structural and biochemical properties of selected residues, this analysis will help them screen and filter out the potentially functionally important residues and direct their experimental design.

Chapter 4: Validation of computational analysis using permutation tests and experimental data

I. Introduction

For the past twenty years, computational analysis has been extensively used in supporting biomedical research and exploring biological data. Its applications range from simple sequence motif matching to complicated simulation of biological systems. Among all computational tools, machine learning has become a flagship technique gaining swift popularity in biological data mining (Baldi and Brunak, 1998). We are one of the first groups to apply machine learning to understand the relationship between protein amino acid sequences and a quantitative functional property (Chapter 2) (Li and Gallin, 2004). My research target is a diverse membrane protein family, voltage-gated potassium channels (VKCs).

VKCs sense and react to the change in voltage difference across the cell membrane. Their operations are gated by the transmembrane voltage (Bezanilla, 2000). The S4 helix, with a characteristic charged residue at every third position, has been demonstrated to be the voltage sensor (Liman et al., 1991; Papazian et al., 1991). While it is still a matter of hot debate, several hypotheses have been proposed to explain how the voltage sensing by S4 is coupled and conveyed to the S5-S6 gating module (Catterall, 1986; Jiang et al., 2003b; Cuello et al., 2004). Besides the S4 domain, a number of residues in S1-S3 helices have also been implicated in voltage sensing of VKCs (Papazian et al., 1995; Tiwari-Woodruff et al., 1997). Although they do not directly sense the change in voltage, the S1-S3 helices appear to modulate the voltage sensitivity of VKCs (Yellen, 1998; Bezanilla, 2000).

Many of these findings were based on functional characterization of VKC wild types and mutants. The design of VKC mutants was based on a combination of prior knowledge of VKC functioning and researcher-specific hypotheses.

I collected 58 VKC sequences and their half activation voltage (V_{50}) values from a voltage-gated potassium channel database (Chapter 2) (Li and Gallin, 2004). Using machine learning techniques, I extracted a computational model that depicts the relationship between amino acid sequences of VKCs and their voltage sensitivities (Chapter 3). The study used an unbiased collection of all collected VKC structural and functional data. The mean absolute error between predicted V_{50} values and V_{50} values determined experimentally in the study is 7.0mV, using a repeated (ten time) ten-fold cross validation (Chapter 3).

Although cross validation is considered a standard operation in evaluating performance of learning classifiers (Witten and Frank, 2000), I wanted to achieve a more objective assessment of the computational analysis. In this chapter, I take two approaches to obtain an independent validation of the predictor. First, I ran permutation tests with the training dataset, which help validate the statistical significance of the computational prediction. Secondly, I collected a group of VKCs wild types and VKC mutants whose V_{50} values were determined using experimental approaches. Using these data as an independent test set, I was able to get an objective estimate of the prediction accuracy of the predictor.

II. Methods

1. Computational model and dataset

A computational model (predictor) was obtained using a KNN classifier combined with a wrapper algorithm for feature selection (Chapter 3). The model predicts the half activation voltage (V_{50}) of a VKC based on its amino acid sequence, with a mean absolute error (MAE) of 7.0mV. The dataset with 54 VKCs that was used to obtain this predictor (Chapter 3) was the original dataset in permutation tests.

2. Permutation test I

This test was designed to test the performance of the final predictor, compared to the null hypothesis that it is no better than random selection of V_{50} values from the training dataset. The class labels (V_{50} values) of VKCs in the original training set were shuffled and then randomly reassigned to these instances. With the predictor that was obtained using a KNN classifier (Chapter 3), these "permutated" instances were reclassified and assigned to a predicted V_{50} value. The MAE and SD between the predicted V_{50} values and their permutated "true" V_{50} values were calculated. The procedure was repeated 10,000 times. Results were compared with the MAE and SD of the predictions with the original training data using the predictor. Student's t-test in both Permutation test I and II was carried out using SigmaPlot 9.0 (Systat Software, Inc).

3. Permutation test II

This test evaluates whether there is significant information linking the sequence of a VKC to its V_{50} value. In this test, I also randomly shuffled the classes of each instance in the original training set. With the new "permutated" training data, I repeated data training using KNN classification combined with the wrapper algorithm, with

identical parameters and settings as in the original training (Chapter 3). This process was repeated 100 times. The best learning performance from each of the 100 repeats were compared with the original best performance (MAE = 7.0mV).

4. Test datasets with experimental data

To obtain a more objective assessment of the predictor, thirteen new VKC sequences and their V_{50} values were collected. Most of them were extracted from the voltage-gated potassium channel database (Chapter 2) (Li and Gallin, 2004). They also included VKCs that were newly cloned and characterized by our lab and other groups (Fry et al., 2004; Salvador-Recatala et al., 2004). These VKC sequences were sent to the predictor for V_{50} prediction. The MAE between predicted V_{50} values and the V_{50} values of these new VKCs determined experimentally was used to estimate the performance of the predictor on new channel sequences.

I also obtained sequences and V_{50} values of six VKC mutants with mutations at one of six residues that were selected by the wrapper algorithm during the original learning process (Chapter 3). These mutants were a part of an alanine scanning mutagenesis experiment by Li-Semrin *et al* (Li-Smerin et al., 2000). The MAE of prediction of V_{50} values of these mutants was also used to evaluate the performance of the predictor.

III. Results

1. Permutation tests

I first performed a simple permutation test to quantify the statistical significance of the computational prediction. A "permutated" dataset was acquired by randomly

shuffling the V_{50} values among these VKC sequences. In 10000 trials, the MAEs between predicted V_{50} value by the predictor and the "permutated" V_{50} value range between 15.5mV and 27.8mV with SDs between 20.6mV and 34.3mV. The performance of the predictor on the original training set (MAE = 7.0mV and SD = 5.1 mV) is significantly better than the mean MAE and SD in this permutation tests ($P < 10^{-10}$) (Table 4.1).

Using the same approach, I obtained another 100 "permutated" datasets. With the same parameters and settings with which I obtained the predictor (Chapter 3), I applied KNN classification combined with the wrapper algorithm for feature selection on each one of these permutated dataset. Different sets of residues were selected for different datasets and the MAEs with the permutated datasets range from 9.9mV to 15.4mV (mean = 13.3mV). Again, the performance of the predictor with the original dataset (MAE = 7.0mV) is significantly better ($P = 2X10^{-10}$) (Table 4.1).

2. Evaluation with VKC wild type data

Thirteen wild type VKCs were input into the predictor for V_{50} predictions. The MAE of these predictions, compared with the experimental data, is 9.7mV (Table 4.2). However, within this test set, a VKC from *Hirudo medicinalis* (Weiss et al., 1999; Salvador-Recatala et al., 2004) generated a prediction error over 27mV (Table 4.2). When this sequence is removed, the MAE of the remaining eleven VKCs is 8.3mV.

Permutation test I	Predictor	P values
Mean MAE 22.3 (15.5 ~ 27.8)	MAE 7.0	$< 10^{-10}$
Mean SD 28.6 (20.6 ~ 34.3)	SD 5.1	$< 10^{-10}$
Permutation test II		
Mean MAE 13.3 (9.9 ~ 15.4)	MAE 7.0	$2X10^{-10}$

Table 4.1: Permutation tests validated the performance of the predictor.

Expression systems	Channels	Published V_{50} (mV)	Predicted V_{50} (mV)
Myocyte	Kv1.2 rabbit	-19.6	-16.5
Neuro-2a	Kv1.4 pig	-17.0	-18.9
RBL	Kv1.4 frog	-26.0	-18.9
Cos7	Kv1.7 mouse	-8.0	-16.5
Xenopus oocytes	Kv1.8 human	3.6	-3.8
Xenopus oocytes	Kv1.10 frog	-11.3	-8.0
CHO	Kv3.3 human	11.0	6.5
HEK	Kv3.3 fish	15.6	6.5
Xenopus oocytes	Kv4 lobster	-19.0	-7.4
HEK	Kv4.2 human	-3.2	-7.4
HEK	Kv4.3 rat	5.0	-7.4
Xenopus oocytes	Kv4 ciona	20.0	-7.4
	MAE	**8.3**	
HEK	*Kv1 leech	8.3	-19.0
	MAE	**9.7**	

Table 4.2: Prediction of independent VKC data using the predictor. The MAE for all thirteen VKCs is 9.7mV, and it is 8.3mV after taking out Kv1 Leech, an evolutionarily distant VKC from any other channel in the training set.

3. Evaluation with VKC mutant data

I also evaluated the predictor by comparing the predicted V_{50} values of six VKC mutants with the experimental data from an Alanine mutagenesis scanning of Kv2.1 rat by Li-Smerin *et al* (Li-Smerin et al., 2000). The comparison was shown in Table 4.3. The MAE between the predictions and data obtained experimentally is 7.5mV, which agrees with the estimated MAE of 7.0mV using cross validation (Chapter 3). Prediction improves significantly (MAE = 1.35mV, n = 2) if only the residue types that VKC mutants were mutated to exist at the same positions in the dataset (Table 4.3).

IV. Discussion

1. Statistical evaluation using permutation tests

A permutation test is a special case of randomization tests. With a small sample of data, it helps generate a distribution for statistical inference. To assess the statistical significance of the computational model with a MAE of 7.0mV (Chapter 3), I first tested the null hypothesis that this predictability occurs simply by chance. In permutation test I. the mean MAE and SD with 10000 permutations of the original data are 22.3mV and 28.6mV, respectively (Table 4.1). Both values are significantly higher ($P < 10^{-10}$) than that with the original training (MAE = 7.0 and SD = 5.1mV), thus rejecting the null hypothesis that the computational prediction of original dataset is generated by chance. In other words, there is almost a 100% possibility ($P < 10^{-10}$) that the predictor does reflect a true relationship between VKC sequences and their V_{50} values.

Ala scan (Kv2.1)	Published V_{50} (mV)	Predicted V_{50} (mV)	Wild type (mV)
L97A	0.6	-7.2	-4.9
*I100A	-7.3	-7.2	-4.9
L117A	-1.6	-7.2	-4.9
*V125A	-4.6	-7.2	-4.9
L135A	1.5	-7.2	-4.9
A154Y	7.0	27.5	-4.9
MAE	**7.5**		

Table 4.3: Published mutant data and predicted V_{50} values. * The residue mutated to Ala at this position exists at the same position in the dataset (Li-Smerin et al., 2000).

The same approach was extended to repeat the full process of KNN classification and feature selection with 100 permutations of the original dataset, using the same parameters used to obtain the computational model (Chapter 3). In this test (Permutation test II), 100 different computational models were generated with 100 different sets of features (residues). Tailored to select a feature set that provides the best learning performance for a specific permutated dataset, the best and worst MAEs among these permutation learning are 9.9mV and 15.4mV, respectively, with a mean MAE of 13.3mV (Table 4.1). The mean MAE is significantly worse than the predictor with the original dataset ($P = 2X10^{-10}$). Since both KNN classification and feature selection process were involved in Permutation test II, each test yielded a "best" model that mathematically correlates a set of residues with "permutated" V_{50} values. It is the fact that the original model (MAE = 7.0mV) significantly outperforms any of the "permutated" models that strongly supports that the original learning is likely to have detected true signals and revealed a valid association between structural elements of VKCs and their activation properties, represented by V_{50} values.

Permutation tests have been widely used in biomedical and other areas including microarray analysis, SNP research, and clinical studies (de Lichtenberg et al., 2004; Listgarten et al., 2004; Potter, 2004). Compared with other statistical analyses, a permutation test works well with small sample sets and it does not require a normal distribution, which many small samples do not have. Some researchers have even proposed that permutation test should be used in all cases (Routledge, 1997).

Both permutation tests clearly indicated the predictor summarizes a legitimate connection between certain residues of VKCs and their V_{50} values. This is particularly

significant in Permutation test II, in which features were reselected from about 300 residues to optimize learning performances with permutations of the original dataset, because, with 300 residues, the likelihood that a few of them correlate with any given set of V_{50} values and thus yield a good prediction by chance is not trivial. The fact that the best MAE in Permutation test II improved to 9.9mV also supports this notion. However, Permutation test II, with a mean MAE of 13.3mV, effectively rejects the possibility that the prediction by the computational model occurs by chance ($P = 2X10^{-10}$). Therefore, the selected residues are likely biologically important in modulating V_{50} values of VKCs; the model for prediction likely contains mechanistic information of VKC activation.

2. Evaluation of predictor using independent experimental data

The goal of machine learning is to extract a model from available data and use this model to accurately predict future instances. In any learning task, ideally, there is a training set for initial training of learning algorithms; there is also a test set that is not "seen" by learning algorithms and exclusively used to evaluate the learning model generated with the training set.

Due to the limited number of data, I did not retain a portion of data as an independent test set when constructing the predictor. Instead, I used a repeated ten-fold cross validation to estimate prediction errors on new instances (Chapter 3). Subsequently, I located another thirteen VKCs with functional characterization including VKCs that were recently cloned (Salvador-Recatala et al., 2004). They formed an independent test set for the predictor. Using the predictor, the MAE of predictions of all thirteen new VKC instances is 9.7mV (Table 4.2), which is higher than what I estimated using a repeated ten-fold cross validation (7.0mV) (Chapter 3).

An independent test set normally produces a more objective evaluation of the learning performance with future instances, provided that the test set has the same data distribution as the training set. However, superposition of the test VKC data on the distance tree of the training data clearly showed an unequal distribution in the sequence space (Figure 4.1). Among the thirteen test VKC data, one VKC is from *Hirudo medicinalis* (Weiss et al., 1999). The prediction error of this VKC using the model is 27.3mV (Table 4.2). The computational model was obtained using a KNN classifier, which classifies an instance based on its "nearest neighbor" in the training set. Having not being trained with VKCs from some species and thus possibly having no "close neighbors" to compare with (Figure 4.1), it is thus expected that the predictor will not perform well with VKCs from them, mostly evolutionarily distant species, such as *Hirudo medicinalis* in the test set. On the other hand, the MAE of the remaining eleven VKC instances in the test set is 8.3mV, indicating that the estimate of prediction accuracy of the predictor using cross validation is reasonable (Chapter 3).

In the training set, all V_{50} values were determined when the channels were expressed in *Xenopus* oocytes (Chapter 3). In the test set, however, I also included VKCs that have V_{50} values determined in other cells, such as HEK and CHO cells (Table 4.2) (Rae and Shepard, 2000; Fry et al., 2004). Although it is known that the experimental V_{50} values of VKCs often vary if they are measured in different cells, the difference is often not significant, as shown by experimental data of several VKCs that have been characterized in both *Xenopus* oocytes and other cells (Table 4.4). Therefore, I believe that the test set can serve as a valid independent test set. In fact, I speculate that a more optimistic estimate will be obtained if all test instances are measured in *Xenopus*

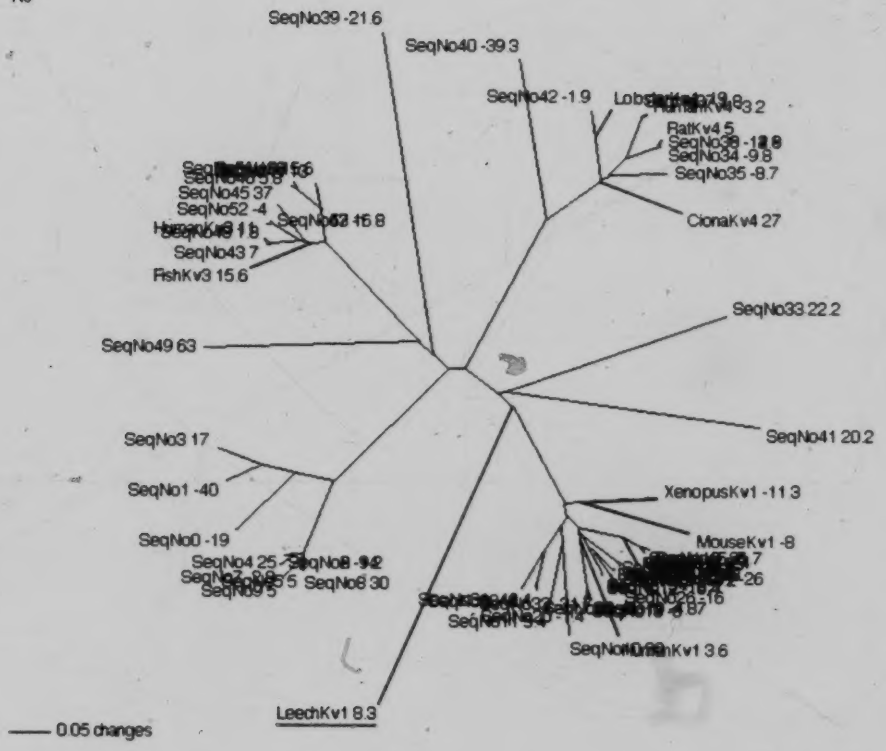


Figure 4.1: Distance tree of training data superimposed by the independent wild type VKC test data. The branches of the independent wild type VKC type data are in red. Most test VKC sequences are clustered with some training data except a VKC from *Hirudo medicinalis* (LeechKv1), which is underlined. The MAE of this VKC is 27.3mV and the MAE of the remaining channels is 8.3mV.

Channels	Xenopus oocytes (mV)	Other cells (mV)	Variations (mV)
Kv1.1 mouse	-26.6	-6.0	20.6
Kv1.5 rabbit	-4.9	-8.7	3.8
Kv1.5 pig	-1.4	-5.0	3.6
Kv1 aplysia	-21.6	-21.1	-0.5

Table 4.4: Variations of V_{50} values measured in different cell hosts.

oocytes because it will remove the variation due to difference cell hosts.

Besides a test set with wild type VKC data, I also compared experimental data from a mutagenesis scanning study by Li-Smerin *et al* with predictions by the predictor (Li-Smerin et al., 2000). Consistent with experimental data, little variation of voltage sensitivity from the wild type (Kv2.1 rat) was predicted for most of these VKC mutants (Table 4.3). These mutants were shown to have little impact on voltage sensitivity if they were mutated to Ala in Kv2.1 (Li-Smerin et al., 2000). One mutant, A154Y of Kv2.1 rat, displayed a large shift of V_{50} of over 10mV (Li-Smerin et al., 2000), and the predictor also predicted a large positive shift in its V_{50} value (Table 1). Although this is the largest margin between the predicted V_{50} and the experimental data, the correct prediction of direction in V_{50} shift by the predictor is encouraging.

It is also expected that the predictor will perform better with certain instances (mutated VKCs, in this case), if the "new" mutated residues (features) exist in one of the sequences in the datasets, and thus have been "seen" by the predictor. Two of the VKC mutants I compared fell into this category and yielded a MAE of 1.35mV between published and predicted V_{50} values (Table 4.3), indicating that prediction accuracy will further improve with more training data available.

Despite using test sets comprising results from VKC mutants and the presumably drastic difference between data distributions of two test sets and the original training set, prediction by the predictor is consistent with experimental results from the study (Table 4.2 and 4.3). This strengthens the conclusion that the estimated prediction error of 7.0mV is close to the true error.

V. Conclusions

I generated a computational model to predict V_{50} values of VKCs based on its amino acid sequences (Chapter 3). The MAE of prediction is 7.0mV, estimated using a repeated ten-fold cross validation. In the present study, I used permutation tests to further validate the model. Two permutation tests demonstrated that the computational model likely reflects the true connection between amino acid residues of VKCs and the activation process. The model was also validated by two independent test sets including wild type VKCs and VKC mutants. Therefore, the model I generated can provide valid information on critical structural elements and functional properties of VKCs.

Chapter 5: General discussion and conclusions

I. Biological data collection and management

1. Information explosion

Traditionally, experimental results acquired from bench work and biological data collected by field observations were recorded in researchers' lab notebooks. There was no centralized storage of biological data. To find relevant information, printed papers and personal communication were the main channels.

In 1965, Dayhoff collected all 65 available protein sequences and organized them in the "Atlas of Protein Sequence and Structure". This was the first protein sequence database, in the form of a book (Dayhoff et al., 1965). In 1982, GenBank, the central repository for DNA sequences, was launched (Benson et al., 2004). Although GenBank quickly outgrew the capacity of floppy disks, DNA sequences and related information from GenBank could still be distributed to researchers in the form of CD-ROMs up to the mid 1990's. Soon, however, the exponentially increasing number of DNA sequences in GenBank, driven by availability of high throughput sequencing techniques and the growth of the human genome project, exceeded the capacity of available low cost portable data storage devices (Benson et al., 2004). As of Oct 2004, there were approximately 43,194,602,655 bases in 38,941,263 sequence records deposited in Genbank (Figure 5.1). Data storage and management have become a critical part of biology now.

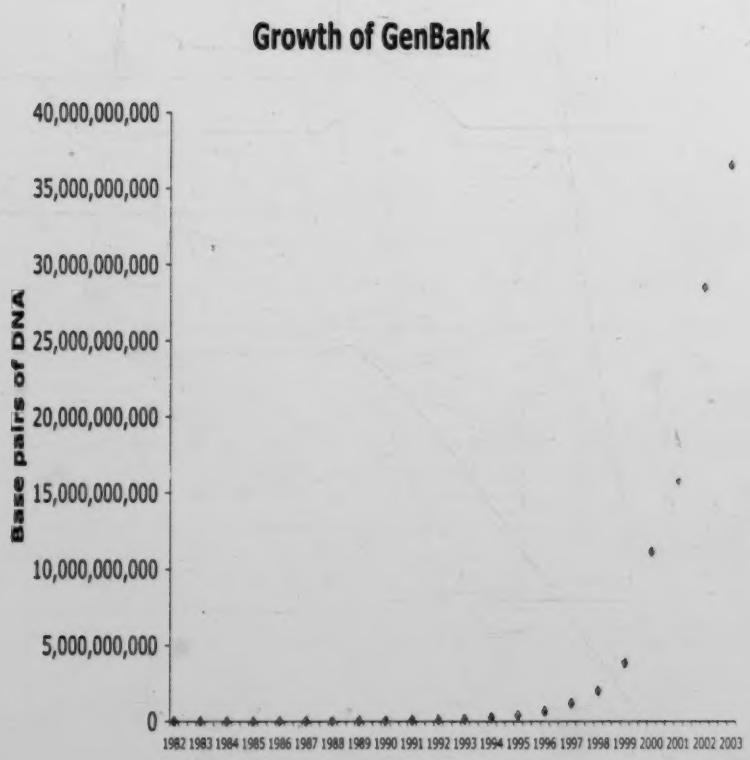


Figure 5.1: The growth of GenBank. The number of base pairs of DNA sequences in GenBank has been growing exponentially.

2. Protein family databases

Besides archival databases, such as GenBank, a variety of specialized biological data such as information on specific protein families are collected and managed in centralized database systems (http://www3.oup.co.uk/nar/database/cap) (Galperin, 2004). One such example is the voltage-gated potassium channel database (VKCDB) (Chapter 2) (Li and Gallin, 2004). Biolgocial database have also extended from simple flat file formats (Benson et al., 2004) to more organized relational databases (Chapter 2) (Li and Gallin, 2004) or object-oriented databases (Harris et al., 2004). These specialized databases often focus on one specific protein family. Most of them contain specialized biological data that can not be found in generic archival databases. As with GenBank, most of these specialized databases are web accessible. Many of them also provide specialized tools that are specifically tailored for this group of proteins to help users browsing, searching, or mining available data based on structural and functional criteria. Despite their relatively small scale, they serve a focused purpose and have a targeted audience; therefore, they have been valuable resources for their specific research communities.

Although data collection for many small protein family databases has been done manually, most databases nowadays are populated with a combined automatic collection and manual curation approach. For VKCDB, specialized eletrophysiological and pharmacological data of voltage-gated potassium channels (VKCs) had to be collected from journal articles. In order to assure accurate information, manual collection was used. The main protein structural and functional information of VKCDB was automatically parsed and populated from GenBank (Benson et al., 2004) and Swissprot (Boeckmann et

al., 2003). The fact that VKCs share high degree of sequence similarity enables us to perform a BLAST search (Altschul et al., 1990) using a seeding VKC sequence to obtain all VKCs that are stored in these archival databases.

By definition, members of most protein families will score high in a BLAST search if a protein sequence from this family is used as the query sequence. Thus, this approach can be readily modified to build and populate databases of other protein families. Attempts to build a generic tool to construct such protein family databases have been published (Horn et al., 2001). They took an essentially similar approach to what was used for construction of VKCDB, using a more generic framework to allow immediate application. Several such databases have been constructed using this approach (Horn et al., 1998; Horn et al., 2001).

Although a generic tool is simple and convenient, a specialized protein family database usually needs more specialized features to make it a better resource than a simple information cluster that can be also obtained from archival databases with some effort. In fact, databases that were constructed using generic tools have all added their own specialty implementations later (Horn et al., 1998; Horn et al., 2001). Every protein family has its own structural and functional characteristics and its own biologically significant analytical information. A good protein family database should provide the specialized knowledge that is pertinent to this family. VKCDB incorporated a number of features that are of great interest to VKC researchers. For example, users can browse VKCDB by Kv family (Chandy, 1991) and electrophysiological parameters to help comparative study of VKCs. A number of VKC-specific information and tools are also available for the specialist users. All these specialized features distinguish VKCDB from

being a simple generic data collection, and provide VKC researchers rich VKC-specific biological data (Chapter 2) (Li and Gallin, 2004). Since it was made public accessible last year, it has received almost 4000 visits, averaging about 25 hits on a typical working day.

3. A comprehensive resource for VKC research

Many improvements can be made to enhance VKCDB as a complete database for VKC research. As a resource for structure-functional study, some VKCDB entries have hyperlinks to protein data bank structure entries (Bourne et al., 2004), but structural data are not stored locally. With recent exciting progress in structural studies of potassium channels (Doyle et al., 1998; Kreusch et al., 1998; Morais Cabral et al., 1998; Bixby et al., 1999; Gulbis et al., 1999; Cushman et al., 2000; Gulbis et al., 2000; Jiang et al., 2001; Sokolova et al., 2001; Jiang et al., 2002a, b; Jiang et al., 2003a; Kuo et al., 2003; Liu and Lin, 2004; Zhou et al., 2004), a centralized storage of available structural data including homology modeling data will greatly facilitate structural comparison and refinement and structure-functional study of VKCs.

Collection of available structural data and subsequent data processing can be carried out automatically. Structural data should be stored separately in the relational database schema with links to VKCDB entries wherever applicable (Chapter 2) (Li and Gallin, 2004). All original structure files will be kept as they are. CGI scripts can be developed to allow users to browse, search, and compare structural information of individual domain across multiple structures. Both original structural data and related information of individual domains will be available on the web.

Besides browsing and searching capacity with structural data, I can also implement structure-related analytical tools on the VKCDB web portal. VADAR, a

structure analysis tool developed by the Wishart group (Willard et al., 2003), for example, can be implemented as a web server at the VKCDB website. Because many of the primary users of VKCDB are biologists without extensive structural analysis or computing experience, the analytical output of VADAR can be parsed and presented in a user-friendly manner with options of selecting only relevant information that is of interest to specific users. For example, users can be allowed to select only residues that are close to certain atoms or likely involved in hydrogen bonding.

If I can make structural information of each individual domain of VKCs available, it will be valuable to implement another useful structure tool, SuperPose (Maiti et al., 2004), on the VKCDB site. SuperPose calculates protein structure superpositions using a modified quaternion approach. It generates structure alignments, PDB coordinates, RMSD statistics, and other analytical results (Maiti et al., 2004). A great deal of our understanding of VKCs comes from comparative studies of the structures and functions of individual domains of VKCs. With SuperPose, users can easily superpose and compare structures of specific domains from structures of different potassium channels in real time, and identify functionally important structural motifs.

Currently, VKCDB contains mainly biological data of wild type VKCs. As we know, site-directed mutagenesis and characterization of protein mutants have provided us with great insight into molecular machinery for protein functioning. Numerous experiments (Heginbotham et al., 1994; Sigworth, 1994; Yellen, 1998; Bezanilla, 2000) have been done with VKC mutants. Therefore, adding information about VKC mutants into VKCDB can further enrich the value of this database. An ion channel mutational database was set up a few years ago, but it is a simple data collection repository and relies

on input from users (http://hoshi-o.physiology.uiowa.edu/cgi-bin/Mutations.pl). I can use their information on VKC mutants as a starting point, and further populate mutational data of VKCs by combining automatic mining of published articles and manual checking. Several tools are available to extract biological information and mutational information from the literature with reasonable accuracies (Donaldson et al., 2003; Horn et al., 2004). These tools can be tested and tuned to fit our needs on VKC data mining. Mutational data should be linked to their wild type cognates in VKCDB. The addition of VKC mutational data will undoubtedly further expand the scope of VKCDB.

Specialized databases, such as VKCDB, have been important resources for studies of protein families. With structural data, mutational data and related data mining tools being integrated into VKCDB, VKCDB can become one of the first specialized databases that serves as a central repository for complete biological information about a specific protein family, including primary sequence information, structural data, functional information on wild type and mutants, as well as a collection of computational tools. VKCDB provides researchers with a fast and ready access to a comprehensive resource for VKC research and helps users mining these data to generate hypotheses that will assist their experimental design.

II. Computational analysis of biological data

1. Machine learning can help

Much of biology is a hypothesis-driven science. Hypotheses are typically derived from a combination of various prior experimental results, observations, and individual researcher's understanding of the problem at hand. In molecular structure-functional

studies, protein sequences are some of the most critical structural data. Sequence comparison using multiple alignments has revealed interesting sequence motifs that have led to exciting findings. In the early 1990's, a highly conserved motif was identified by aligning sequences from a number of potassium channels including VKCs, ligand-gated potassium channels, and other types of potassium channels. This five-residue motif, TXGXG, is conserved among all aligned potassium channels, and was immediately hypothesized to be the signature motif of potassium channels (Heginbotham et al., 1994). Characterization of mutants within this motif confirmed that it is the motif that determines the potassium selectivity of potassium channels (Heginbotham et al., 1992; Heginbotham et al., 1994). Similar analyses of sequence data as well as functional characterization have helped identify other important structural elements of VKCs, including the voltage sensor, the S4 helix (Liman et al., 1991; Papazian et al., 1991).

Although successful hypotheses can be formulated by inspection of sequence alignments, this approach is limited by the complexity of available data. For residues that are not well conserved but play significant roles in functional diversity, it becomes difficult to identify them and hypothesize on their roles based on inspection of multiple alignments of sequences. If the structure-functional relationship is to be studied, functional characterization has to be included to obtain a detailed map of structure-functional association, making the dataset too complex to understand by simple inspection. On the other hand, machine learning, a data mining tool that is built to identify rules of feature-class relationships (Mitchell, 1997), such as structure elements and functions, is an appropriate technique for tackling this type of problems.

Machine learning generalizes a given data set and extracts a model that describes how attributes of these data determine their classifications. Using data extracted from VKCDB, I have successfully applied machine learning techniques and generated a classifier that predicts the half activation voltage (V_{50}) values with a mean absolute error of 7.0mV, using only the amino acid sequence of a VKC (Chapter 3).

2. Quality and quantity of training data

Learned from the training data, the final classifier is limited by the quality and quantity of the training data. The dataset contains only 58 VKC sequences and their V_{50} values. Compared with a typical dataset for machine learning, the dataset has a very limited number of instances. A dataset with more instances will undoubtedly help improve the learning. I tested a dataset of 44 instances earlier using the same approach and the best learning performance was 11.0mV. The dataset of 54 instances improved the learning performance to 7.0mV (Chapter 3). As more and more VKCs are identified and characterized each year, an update of the dataset by collecting more VKC sequences with characterized V_{50} values is likely the simplest and the best method to improve the learning performance. Besides manual browsing, automatic literature searching can be tested, using text mining tools to extract literature that contains functional information of VKCs to help collect more V_{50} values (Donaldson et al., 2003; Horn et al., 2004).

In addition, the small size of the dataset implies that the training data are not likely to include the same number of channels from all species. Limited by experimental techniques and varying abundance of different VKCs, the training data I collected are not a well-balanced sample set; most of the training data come from human and model organisms (Chapter 2) (Li and Gallin, 2004). A distance tree of the channels used in

constructing the predictor illustrates the uneven representation of channels in sequence space (Figure 5.2).

The final predictor was built with a KNN classifier (Chapter 3). In KNN classification, a close "neighbor" from the training set will be used as a template to classify a new instance. If the training data are not evenly distributed in the instance space, some areas contain fewer instances with larger empty space than others, as shown in the distance tree of the training data (Figure 5.2). Evidently, instances that are in these sparse areas will likely not be accurately classified, since they do not have "close" neighbors. In fact, a VKC in the independent test set falls into such category (Figure 4.1), and the predictor performed poorly (MAE = 27.3mV), while a better prediction was achieved with the rest of test data (MAE = 8.3mV) (Table 4.2) (Chapter 4). Therefore, a more phylogenetically diverse selection of channels should improve performance.

Data accuracy of the dataset is another issue. Because the V_{50} values in the datasets were measured in different labs over a span of two decades (Chapter 2) (Li and Gallin, 2004), there are likely quantitative errors of varying magnitude in this dataset. The complicated nature of voltage sensing, the unstandarized experimental procedures and human errors can all lead to noisy data. I found that several labs reported different V_{50} values for a single VKC (Stuhmer et al., 1989; Schroter et al., 1991; Rettig et al., 1992; Scholle et al., 2000). In fact, the V_{50} values of a VKC that were determined in different experiments by the same lab were still 5mV apart (Monks et al., 1999; Hong and Miller, 2000). Since many factors and conditions are involved in an eletrophysiological experiment, a well-standardized experimental technique is not likely to emerge in the near future. The reliable approach is to verify these data by

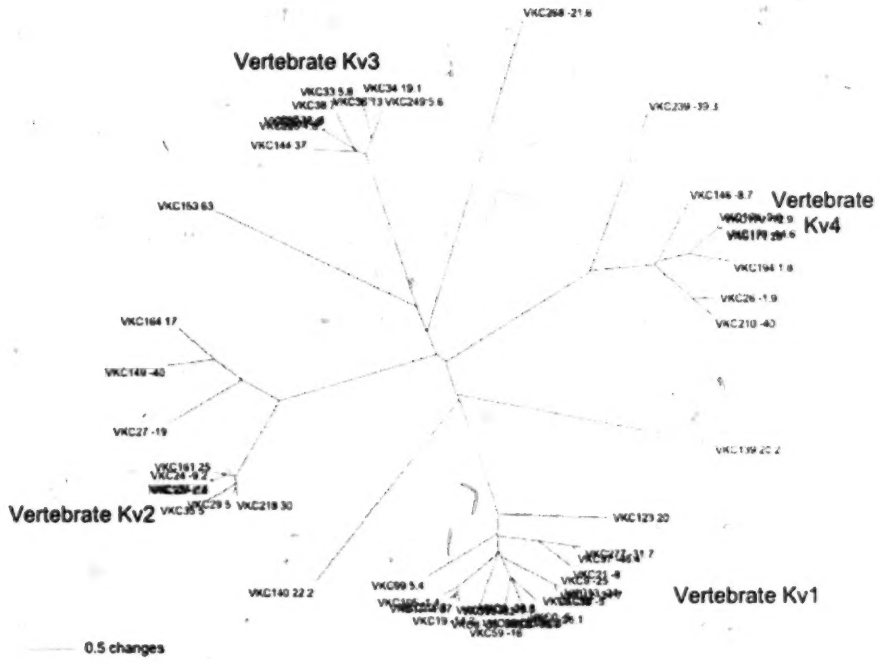


Figure 5.2: The distance tree of all training data. It shows an uneven distribution samples in sequence space.

independently repeated experiments. Using the average of V_{50} values of the same channel from multiple experiments, which is expected to lower the variance and obtain a V_{50} value that is close to the true value, should alleviate this problem. It is also possible that the MAE of 7.0mV that we obtained is the best result we can achieve with this dataset if the V_{50} values in the training set are imprecise.

I used a heuristic and intuitive approach to identify four possible outliers. They were selected based on their negative effect on the overall learning performance (Chapter 3). Misclassification is commonly seen with outliers. Two of the four selected outliers are Kv4 channels (Vogalis et al., 1995; Holmqvist et al., 2002a), and their fast-inactivating nature could lead to inaccurate V_{50} measurement. On the other hand, due to the drastic difference in abundance of different VKCs and experimental preferences, the dataset is unbalanced. A few hundred VKCs have been found and they carry out a variety of functions in different organisms, but VKCs from some species are less abundant and naturally under-represented, which could result in them being selected as outliers.

One possible solution to the problem of scarce VKC data is to incorporate mutational data in the dataset. I did a few tests with some mutational data from several mutagenesis scanning studies (Li-Smerin et al., 2000; Minor et al., 2000; Yifrach and MacKinnon, 2002), and they did improve learning performances to some extent. However, this could be problematic because mutagenesis scanning was done with one single channel. Dozens of mutants from one channel added into a dataset with less than 60 instances will significantly over-represent this branch of VKCs. The final result will be overly biased toward this channel and thus difficult to generalize. Designed by researchers, these mutants may also distort the natural mechanism of wild type VKC

functioning and mislead us in VKC study. Therefore, extra caution is needed in mining VKC mutant data. However, with a reasonable number of mutants from a few channels, I can certainly test for possible misleading effects by separately training data from mutants of individual channels. A model that can best describe structure-functional association of a specific channel might be obtained, if there is enough variation in structures and functions among these mutants.

3. Enrichment of training data

The functional form of a protein is a three-dimensional structure, and the connection between primary sequences and their 3D organization is still far from clear. Precise functional prediction using only amino acid sequence is thus a doubly challenging problem. To achieve better functional prediction, I can also incorporate 3D structural information into the dataset. Parameters extracted from homology models based on solved structures of VKCs can be included in the training set, including accessible surface areas, secondary structures, hydrogen bonding information, salt bridge formation and other parameters.

Besides 3D structural data from solved structures, some biochemical properties and secondary structures of VKC residues in the dataset could also be included. These parameters, including hydrophobic moment, polarity, and secondary structure prediction, can be approximately calculated using computational tools (Eisenberg et al., 1982; Krogh et al., 2001). This can be achieved by a Perl script that allows automatic submission of query sequences to and parsing of results from related web servers.

This resulting new dataset would contain not only quantitatively more data but also data with structural information. It would present significantly more informative

structural attributes for machine learning modules than primary sequence alone. It can thus potentially compensate for the limited size of the dataset.

4. Learning algorithms

In Chapter 3, several different learning algorithms were trained. The KNN classifier (k = 1) yielded the best result (Chapter 3). As one of the "lazy" learning algorithms, the KNN classifier simply stores all the training data during training and leaves the computational work to the classification stage, when it compares the new instance with the training data and assigns the new instance the class of its closest neighbor (Mitchell, 1997).

Determination of voltage sensitivity of VKCs is a highly coordinated process (Bezanilla, 2000). Many residues are involved. Some of them are critical in voltage sensing and others may play a modulating role (Bezanilla, 2000). Extensive interaction among some residues and their interaction with membrane lipid molecules have been reported (Larsson et al., 1996; Tiwari-Woodruff et al., 1997; Alvis et al., 2003; Williamson et al., 2003). It is possible that a single model can not include all the factors and accurately describe the voltage sensitivity of a VKC.

The KNN classifier is a simple learning method. However, it often works well, especially for complex problems that can not be explained by a simple model. Unlike some other learning algorithms, the KNN classifier does not generate a single description that fits all training data. In a sense, it is almost equivalent to having each training instance as a model and classifying new instances by comparing it with all these models, which is possibly part of the reason the KNN classification performed well with the dataset.

While KNN classifier generated the best predictor, the Naïve Bayes classifier and the decision tree did not perform as well (Chapter 3). Evidently, different learning algorithms have different intrinsic weaknesses. For a given dataset, different algorithms might perform differently, as evidenced by the results (Chapter 3). It is thus possible that prediction accuracy may further improve if a different learning algorithm is trained.

Developed by Vapnik, based on his learning theoretical study (Vapnik, 1995), the support vector machine approach (SVM) is attracting more and more attention, especially in its applications to mining biological data (Byvatov and Schneider, 2003; Bhasin and Raghava, 2004; Mika and Rost, 2004; Yan et al., 2004). Instead of trying to fit all training data, SVM relies on only a small portion of the data that are instrumental in classification, the so-called support vectors, to classify instances (Vapnik, 1995). Therefore, SVM is relatively insensitive to overfitting, a universal problem in machine learning, particularly for small datasets (Vapnik, 1995).

However, since all features are required to be continuous real numbers in SVM learning, the amino acid sequences, features in the dataset, have to be recoded into real numbers that reflect their biochemical relationships among each other, which could complicate the problem. Nevertheless, SVM has been successfully used in many biological applications, so it can be tested with the dataset if a good method is located that can meaningfully recode amino acid sequences into real numbers.

5. Feature selection

The curse of dimensionality led to development of several feature selection techniques (Blum and Langley, 1997). Typically, there are a few hundred or more features in a protein with each amino acid residue as a feature. Because of experimental

constraints, often only a few dozens of proteins of interest have been sequenced and functionally characterized for comparative study of the structure-functional relationship. Reducing the number of features becomes essential.

Some statistical techniques have been used in reducing dimension of data, such as principle component analysis (PCA) (Jolliffe, 2002). However, PCA does not include classification information and is thus "unsupervised" in nature. Consequently, it may not optimize the learning model. In addition, PCA does not necessarily select individual relevant features. Instead, it generates "superfeatures" that consist of multiple features (Jolliffe, 2002), which may not be appropriate if identification of important individual features (residues) is critical, which is the goal of many analyses with protein datasets.

I tested two "supervised" techniques that incorporate information of classification in their analyses: the filter and wrapper algorithms (Chapter 3). The wrapper algorithm outperformed the filter algorithm with the VKC dataset. I used a forward stepwise selection of features, adding one feature at each round of training (Figure 1.8). Due to the computational complexity, the feature space could not be exhaustively searched. I applied a greedy search and advanced only a portion of the feature sets at each round (Chapter 3). A "residue swapping" test was done at the end and the best feature set remained unchanged (Chapter 3). Although it is likely I located the true global optimum, the possibility of having reached only a local optimum could not be formally excluded.

Despite its relative simplicity, the wrapper algorithm has been successfully applied to many biological data mining problems (Jelonek and Stefanowski, 1997; Degroeve et al., 2002; Inza et al., 2004; Mao, 2004). Some studies indicated that it performs as well as other, more complicated, techniques (Weber et al., 2004). The

permutation tests and experimental data from other groups also indirectly supported the conclusion that residues selected by the wrapper algorithm in the present study are likely functionally important in determining voltage sensitivity of VKCs (Chapter 4).

6. Evaluation of learning

The number of training data was small, so it was difficult to hold out a portion of data for evaluation. Instead, I used a repeated ten-fold cross validation (Chapter 3). Ten-fold cross validation is the *de facto* standard in evaluating learning performance (Witten and Frank, 2000). I used a single ten-fold cross validation in the earlier trials. Despite using the average score of ten tests with each of the ten-folds, the final results did not seem to converge. I obtained slightly different results from each run. Evidently, the instability came from the bias on how training data were divided into ten groups. I then chose to use a ten times ten-fold cross validation, which involved random dividing of training data ten times to minimize the bias. This approach generated consistent outputs (Chapter 3).

As with any typical protein dataset, the class distribution of the training data is likely skewed. A better schema for cross validation is to create cross validation partitions in a way that the proportion of each class remains the same in each partition (Braga-Neto and Dougherty, 2004). For the continuous V_{50} values in the dataset, I can achieve this by approximately dividing training instances into several groups based on the magnitude of their V_{50} values. Bootstrapping can also be tested to obtain a better estimate of learning performance (Zhao et al., 2001; Draghici et al., 2003).

An independent evaluation of the classifier was done by comparing predicted V_{50} values using the classifier with experimental data that were obtained independently. Both

a wild type VKC test set I collected and a VKC mutant test set by Li-Smerin *et al* (Li-Smerin et al., 2000) showed consistent prediction using the predictor, with MAEs that are close to the estimated MAE by cross validation (Table 4.2 and 4.3) (Chapter 4). Although it is certain that the distribution of wild type VKCs and VKC mutants is rather different from the training data, as independent test data that were not used in training, the successful prediction of these instances gave us a reasonably objective assessment of the present predictor. They indicated the prediction accuracy of future instances using the predictor should be close to the MAE I estimated using cross validation, 7.0mV (Chapter 3).

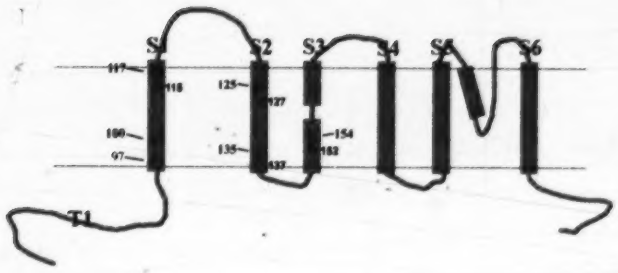
7. Biological significance of identified residues

During the learning process, the wrapper algorithm identified six informative residues that are likely to be involved in modulating the voltage sensitivity of VKCs (Chapter 3). Within the dataset, the variation of the six selected residues are mostly limited to nonpolar, hydrophobic residues including Ile, Leu, Val, Phe, and Ala, with the exception of His at residue 117 for all Kv3 (*Shaw*) channels (Jan and Jan, 1997b) (Table 5.1). Some residues, such as 117, 125, and 154, can potentially interact with residues from S3, S5 and S6, assuming the KvAP structure model is correct (Jiang et al., 2003a). A number of residues from S5 and S6 are within 5Å of position 117 on S1 and position 125 on S3 (Figure 5.3B). Some residues at the S3 turn loop and S3b helix also are in close proximity (<5Å) to position 154 in the C terminus of S3a (Figure 5.3B), calculated using MolMol (Koradi et al., 1996). Although no hydrogen bond was identified among the selected residues and their "close" neighbors, hydrophobic interactions can certainly occur among some of them.

Positions	Residue types in the dataset
97	*C, F, I, L, V, *Y
100	A, *F, *G, I, *L, *T, V
117	H, I, L, V
125	A, *C, *F, I, L, T, V
135	I, L, *T, V
154	A, *C, *F, I, L, *M, V

Table 5.1: Amino acid residues at selected positions in the dataset.

* These residues appear in only one or two VKCs in the dataset of 58 VKCs.



A

B

Figure 5.3: Residues selected as informative features by a wrapper algorithm and mapped onto VKC structures.

A: The six residues selected by the wrapper algorithm were approximately mapped onto a schematic of a VKC. All six of them are located in S1-S3 region, a "non-critical" and "modulating" region in voltage sensing. Residue 117, 125, 135 and 154 are all exactly two residues away from a relatively conserved negatively charged residue, marked by asterisks and underlined indeces (115, 127, 137 and 152).

B: Selected residues were mapped onto the structure of KvAP (Jiang et al., 2003a) in ball-stick mode. Residues that are located within 0.5nm from selected residues are also shown in green CPK mode. This is the cytoplasmic view down the axis of S5 and S6. This figure was prepared using MolMol (Koradi et al., 1996).

Hydrophobic interaction has been known to play important roles in many physiological processes. One of the examples is the N type inactivation of VKCs, in which a hydrophobic fragment and its nonspecific hydrophobic interaction were shown to mediate the fast inactivation of some VKCs (Murrell-Lagnado and Aldrich, 1993).

Many residues of VKCs that are responsible for voltage sensing and selective ion permeation are charged or polar amino acids, generating relatively strong ionic interaction (Yellen, 1998). Variations of residues involved in strong ionic and bonding interactions often lead to, drastic variation or inactivation in function (Yellen, 1998). Hydrophobic interactions between nonpolar residues, on the other hand, are often energetically of lower magnitude and thus have quantitatively smaller effects in the overall function of protein. They can potentially generate interactions with a continuous range of directions and magnitudes. Variations among involved residues are not expected to cause functional disruption but fine-tune the directional and quantitative characteristics. They are likely to play "secondary" roles in VKC functioning and help tuning and shaping the sensitivity of different functional properties. The nonpolar hydrophobic features of identified potential voltage sensitive residues are consistent with their roles in modulating the targeted functional feature, the voltage sensitivity of VKCs.

I mapped these residues onto their homologous positions in the structure of KvAP (Figure 5.3B) (Jiang et al., 2003a). Based on their positions, some of them could be in contact with lipid molecules. It has been demonstrated that the lipid bilayer has an influence on the orientation and positioning of potassium channels and other membrane proteins, possibly by interaction between lipid molecules and protein residues (Alvis et al., 2003; Williamson et al., 2003). Certain functions of potassium channel will likely be

modulated by this interaction (Alvis et al., 2003; Williamson et al., 2003).

Interestingly, four of the six selected residues are exactly two residues away from relatively conserved negatively charged residues in primary sequence (Residue E45, D62, D72, and E93 in KvAP) (Jiang et al., 2003a) (Figure 5.3A), which means that they are on opposite faces of the predicted helices (Figure 5.4). All these charged residues have been known to closely relate to voltage sensing and gating, mostly by interacting with residues with positive charges from the core structural domains (S4-S6) to assist activation, inactivation, or the transition process (Yellen, 1998; Laine et al., 2004). Because of their proximity, it is possible that the selected residues modulate these "critical" residues and their influences on voltage sensitivity by a weak interaction such as hydrophobic interaction, whose magnitude varies based on the physical structures of and the distance between residues that are involved. Further study is needed to clarify this issue.

III. Conclusions

Most biologists work with one or a few specific protein families. The amount of sequence and functional data on members of protein families is increasing rapidly with the growing application of high throughput experimental methods to functional protein studies. Computational data management and data mining become necessary for biologists.

To utilize available sequence data and functional data of voltage-gated potassium channels in comparative study, a voltage-gated potassium channel database (VKCDB) was constructed (Chapter 2) (Li and Gallin, 2004). It was populated through iterative BLAST search of GenBank and Swissprot. Annotations for all voltage-gated potassium

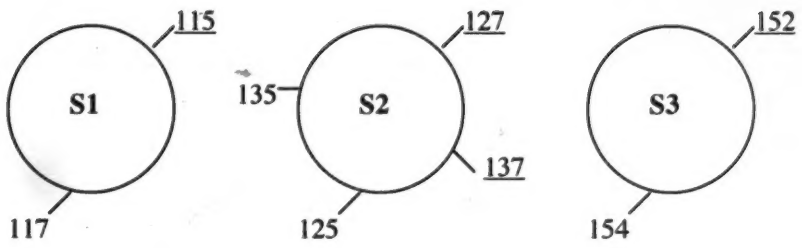


Figure 5.4: Selected residues and neighboring charged residues on a schematic axial view of the S1-S3 helices. Among the six selected residues, four of them are two residues away from a negatively charged residue in primary sequence. When they are mapped onto the S1-S3 helices, the selected residues (117, 125, 135, and 154) and their neighboring charged residues (115, 127, 137, and 152) are approximately located on the opposite faces of S1-S3 helices.

channels were selectively parsed and integrated into VKCDB. Electrophysiological and pharmacological data for the channels were collected from published journal articles. Transmembrane domain predictions by TMHMM and PHD are included for each VKCDB entry. Multiple sequence alignments of conserved domains of channels of the four Kv families and the KCNQ family were also included. VKCDB can be browsed and searched using a set of functionally relevant categories. Since it was made available on the web, it has become an important resource for potassium channel research community.

Using collected amino acid sequences and electrophysiological data, several machine learning algorithms were trained to produce a predictor that can predict the half activation voltage, one of the central electrophysiological parameters, of a given VKC based on only its amino acid sequence with a good accuracy (MAE = 7.0mV). Prediction was verified by permutation tests and independent experimental data from several research groups (Table 4.2 and 4.3) (Chapter 4). During the process, a number of residues were shown to be correlated with quantitative features of VKCs. They are thus likely functionally critical in VKC activation. VKC mutants have been made based on computational predictions, and their functional characterization is underway to further study their roles in VKC functioning.

The approach I used to build VKCDB and the methodology I used for computational analysis of structure-functional relationship of VKCs are not specifically tailored for VKCs. They can be easily generalized and modified for studies of other protein families.

References

Abdul, M., and N. Hoosein. 2002a. Voltage-gated potassium ion channels in colon cancer. *Oncol. Rep.* 9:961-964.

Abdul, M., and N. Hoosein. 2002b. Voltage-gated sodium ion channels in prostate cancer: expression and activity. *Anticancer Res.* 22:1727-1730.

Adams, E.N., III. 1972. Consensus techniques and the comparison of taxonomic trees. *Systematic Zoology.* 21:390-397.

Alberts, B., A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter. 2002. Molecular Biology of the Cell, 4th edition. Garland Science Publishing, New York.

Almuallim, H., and T.G. Dietterich. 1991. Learning with many irrelevant features. *In* Ninth National Conference on Artificial Intelligence. MIT Press, Anaheim, CA. 547-552.

Altschul, S.F. 1991. Amino acid substitution matrices from an information theoretic perspective. *J. Mol. Biol.* 219:555-565.

Altschul, S.F., W. Gish, W. Miller, E.W. Myers, and D.J. Lipman. 1990. Basic local alignment search tool. *J. Mol. Biol.* 215:403-410.

Altschul, S.F., T.L. Madden, A.A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and D.J. Lipman. 1997. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.* 25:3389-3402.

Alvis, S.J., I.M. Williamson, J.M. East, and A.G. Lee. 2003. Interactions of anionic phospholipids and phosphatidylethanolamine with the potassium channel KcsA. *Biophys J.* 85:3828-3838.

Armstrong, C.M. 1981. Sodium channels and gating currents. *Physiol. Rev.* 61:644-683.

Ashcroft, F.M. 2000. Ion Channels and Disease: Channelopathies. Academic Press. 481 pp.

Baldi, P., and S. Brunak. 1998. Bioinformatics: The Machine Learning Approach. The MIT Press, Cambridge, MA.

Benson, D.A., I. Karsch-Mizrachi, D.J. Lipman, J. Ostell, and D.L. Wheeler. 2004. GenBank: update. *Nucleic Acids Res.* 32 Database issue:D23-26.

Bezanilla, F. 2000. The voltage sensor in voltage-dependent ion channels. *Physiol Rev.* 80:555-592.

Bhasin, M., and G.P. Raghava. 2004. ESLpred: SVM-based method for subcellular localization of eukaryotic proteins using dipeptide composition and PSI-BLAST. *Nucleic Acids Res.* 32:W414-419.

Bixby, K.A., M.H. Nanao, N.V. Shen, A. Kreusch, H. Bellamy, P.J. Pfaffinger, and S. Choe. 1999. Zn2+-binding and molecular determinants of tetramerization in voltage-gated K+ channels. *Nat. Struct. Biol.* 6:38-43.

Blum, A.L., and P. Langley. 1997. Selection of relevant features and examples in machine learning. *Artificial Intelligence.* 97:245-271.

Blumer, A., A. Ehrenfeucht, D. Haussler, and M.K. Warmuth. 1989. Learnability and the Vapnik-Chervonenkis Dimension. *J Acm.* 36:929-965.

Boeckmann, B., A. Bairoch, R. Apweiler, M.C. Blatter, A. Estreicher, E. Gasteiger, M.J. Martin, K. Michoud, C. O'Donovan, I. Phan, S. Pilbout, and M. Schneider. 2003. The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003. *Nucleic Acids Res.* 31:365-370.

Bose, I., and R.K. Mahapatra. 2001. Business data mining - a machine learning perspective. *Inform Manage.* 39:211-225.

Bourne, P.E., K.J. Addess, W.F. Bluhm, L. Chen, N. Deshpande, Z. Feng, W. Fleri, R. Green, J.C. Merino-Ott, W. Townsend-Merino, H. Weissig, J. Westbrook, and H.M. Berman. 2004. The distribution and query systems of the RCSB Protein Data Bank. *Nucleic Acids Res.* 32 Database issue:D223-225.

Braga-Neto, U.M., and E.R. Dougherty. 2004. Is cross-validation valid for small-sample microarray classification? *Bioinformatics.* 20:374-380.

Breiman, L. 2001. Statistical modeling: the two cultures. *Statistical Science.* 16:199-215.

Byvatov, E., and G. Schneider. 2003. Support vector machine applications in bioinformatics. *Appl Bioinformatics.* 2:67-77.

Cabello, D., S. Barro, J.M. Salceda, R. Ruiz, and J. Mira. 1991. Fuzzy K-nearest neighbor classifiers for ventricular arrhythmia detection. *Int. J. Biomed. Comput.* 27:77-93.

Cachero, T.G., A.D. Morielli, and E.G. Peralta. 1998. The small GTP-binding protein RhoA regulates a delayed rectifier potassium channel. *Cell.* 93:1077-1085.

Caprini, M., S. Ferroni, R. Planells-Cases, J. Rueda, C. Rapisarda, A. Ferrer-Montiel, and M. Montal. 2001. Structural compatibility between the putative voltage sensor of voltage-gated K+ channels and the prokaryotic KcsA channel. *J Biol Chem.* 276:21070-21076.

Cardie, C. 1993. Using decision trees to improve case-based learning. *In* Tenth International Conference on Machine Learning. Morgan Kaufmann Publishers, Inc., San Meteo, CA. 25-32.

Catterall, W.A. 1986. Molecular properties of voltage-sensitive sodium channels. *Annu. Rev. Biochem.* 55:953-985.

Cedeno, W., and D.K. Agrafiotis. 2003. Using particle swarms for the development of QSAR models based on K-nearest neighbor and kernel regression. *J. Comput. Aided Mol. Des.* 17:255-263.

Cha, A., and F. Bezanilla. 1997. Characterizing voltage-dependent conformational changes in the Shaker K+ channel with fluorescence. *Neuron.* 19:1127-1140.

Chandy, K.G. 1991. Simplified gene nomenclature. *Nature.* 352:26.

Choe, S. 2002. Potassium channel structures. *Nat Rev Neurosci.* 3:115-121.

Chung, S.H., and S. Kuyucak. 2002. Ion channels: recent progress and prospects. *Eur. Biophys. J.* 31:283-293.

Collins, F.S., M. Morgan, and A. Patrinos. 2003. The Human Genome Project: lessons from large-scale biology. *Science.* 300:286-290.

Comu, S., M. Giuliani, and V. Narayanan. 1996. Episodic ataxia and myokymia syndrome: a new mutation of potassium channel gene Kv1.1. *Ann. Neurol.* 40:684-687.

Cooper, E.C. 2001. Potassium channels: how genetic studies of epileptic syndromes open paths to new therapeutic targets and drugs. *Epilepsia.* 42 Suppl 5:49-54.

Cuello, L.G., D.M. Cortes, and E. Perozo. 2004. Molecular architecture of the KvAP voltage-dependent K+ channel in a lipid bilayer. *Science.* 306:491-495.

Cushman, S.J., M.H. Nanao, A.W. Jahng, D. DeRubeis, S. Choe, and P.J. Pfaffinger. 2000. Voltage dependent activation of potassium channels is coupled to T1 domain structure. *Nat. Struct. Biol.* 7:403-407.

Dayhoff, M.O., R.V. Eck, M.A. Chang, and M.R. Sochard. 1965. Atlas of Protein Sequence and Structure. Natl. Biolmed. Res. Found., Washington, D.C.

de Lichtenberg, U., L.J. Jensen, A. Fausboll, T.S. Jensen, P. Bork, and S. Brunak. 2004. Comparison of computational methods for the identification of cell cycle regulated genes. *Bioinformatics.*

Degroeve, S., B. De Baets, Y. Van De Peer, and P. Rouze. 2002. Feature subset selection for splice site prediction. *Bioinformatics.* 18 Suppl 2:S75-S83.

Donaldson, I., J. Martin, B. de Bruijn, C. Wolting, V. Lay, B. Tuekam, S. Zhang, B. Baskin, G.D. Bader, K. Michalickova, T. Pawson, and C.W. Hogue. 2003. PreBIND and Textomy--mining the biomedical literature for protein-protein interactions using a support vector machine. *BMC Bioinformatics.* 4:11.

Doyle, D.A., J. Morais Cabral, R.A. Pfuetzner, A. Kuo, J.M. Gulbis, S.L. Cohen, B.T. Chait, and R. MacKinnon. 1998. The structure of the potassium channel: molecular basis of K+ conduction and selectivity. *Science.* 280:69-77.

Draghici, S., O. Kulaeva, B. Hoff, A. Petrov, S. Shams, and M.A. Tainsky. 2003. Noise sampling method: an ANOVA approach allowing robust selection of differentially regulated genes measured by DNA microarrays. *Bioinformatics.* 19:1348-1359.

Eisenberg, D., R.M. Weiss, and T.C. Terwilliger. 1982. The helical hydrophobic moment: a measure of the amphiphilicity of a helix. *Nature.* 299:371-374.

Eldstrom, J., K.W. Doerksen, D.F. Steele, and D. Fedida. 2002. N-terminal PDZ-binding domain in Kv1 potassium channels. *FEBS Lett.* 531:529-537.

Fleishman, S.J., O. Yifrach, and N. Ben-Tal. 2004. An evolutionarily conserved network of amino acids mediates gating in voltage-dependent potassium channels. *J Mol Biol.* 340:307-318.

Fry, M., R.A. Maue, and F. Moody-Corbett. 2004. Properties of Xenopus Kv1.10 channels expressed in HEK293 cells. *J. Neurobiol.* 60:227-235.

Galperin, M.Y. 2004. The Molecular Biology Database Collection: 2004 update. *Nucleic Acids Res.* 32 Database issue:D3-22.

Grissmer, S., B. Dethlefs, J.J. Wasmuth, A.L. Goldin, G.A. Gutman, M.D. Cahalan, and K.G. Chandy. 1990. Expression and chromosomal localization of a lymphocyte K+ channel gene. *Proc Natl Acad Sci U S A.* 87:9411-9415.

Gulbis, J.M., S. Mann, and R. MacKinnon. 1999. Structure of a voltage-dependent K+ channel beta subunit. *Cell.* 97:943-952.

Gulbis, J.M., M. Zhou, S. Mann, and R. MacKinnon. 2000. Structure of the cytoplasmic beta subunit-T1 assembly of voltage-dependent K+ channels. *Science.* 289:123-127.

Harris, T.W., N. Chen, F. Cunningham, M. Tello-Ruiz, I. Antoshechkin, C. Bastiani, T. Bieri, D. Blasiar, K. Bradnam, J. Chan, C.K. Chen, W.J. Chen, P. Davis, E. Kenny, R. Kishore, D. Lawson, R. Lee, H.M. Muller, C. Nakamura, P. Ozersky, A. Petcherski, A. Rogers, A. Sabo, E.M. Schwarz, K. Van Auken, Q. Wang, R. Durbin, J. Spieth, P.W. Sternberg, and L.D. Stein. 2004. WormBase: a multi-species resource for nematode biology and genomics. *Nucleic Acids Res.* 32 Database issue:D411-417.

Hartmann, H.A., G.E. Kirsch, J.A. Drewe, M. Taglialatela, R.H. Joho, and A.M. Brown. 1991. Exchange of conduction pathways between two related K+ channels. *Science.* 251:942-944.

Hayes, W.S., and M. Borodovsky. 1998. How to interpret an anonymous bacterial genome: machine learning approach to gene identification. *Genome Res.* 8:1154-1171.

Heginbotham, L., T. Abramson, and R. MacKinnon. 1992. A functional connection between the pores of distantly related ion channels as revealed by mutant K+ channels. *Science.* 258:1152-1155.

Heginbotham, L., Z. Lu, T. Abramson, and R. MacKinnon. 1994. Mutations in the K+ channel signature sequence. *Biophys. J.* 66:1061-1067.

Heginbotham, L., and R. MacKinnon. 1992. The aromatic binding site for tetraethylammonium ion on potassium channels. *Neuron.* 8:483-491.

Henikoff, S., and J.G. Henikoff. 1992. Amino acid substitution matrices from protein blocks. *Proc. Natl. Acad. Sci. USA*. 89:10915-10919.

Herbrich, R. 2002. Learning Kernel Classifiers Theory and Algorithms. The MIT Press, Cambridge, MA.

Higgins, D.G., J.D. Thompson, and T.J. Gibson. 1996. Using CLUSTAL for multiple sequence alignments. *Methods Enzymol.* 266:383-402.

Hille, B. 1970. Ionic channels in nerve membranes. *Prog. Biophys. Mol. Biol.* 21:1-32.

Hille, B. 2001. Ion channels of excitable membranes, 3rd edition. Sinauer, Sunderland, MA. xviii, 814 , [818] of plates pp.

Hodgkin, A.L., and A.F. Huxley. 1952a. The components of membrane conductance in the giant axon of Loligo. *J Physiol.* 116:473-496.

Hodgkin, A.L., and A.F. Huxley. 1952b. Currents carried by sodium and potassium ions through the membrane of the giant axon of Loligo. *J Physiol.* 116:449-472.

Hodgkin, A.L., and A.F. Huxley. 1952c. The dual effect of membrane potential on sodium conductance in the giant axon of Loligo. *J Physiol.* 116:497-506.

Hodgkin, A.L., and A.F. Huxley. 1952d. A quantitative description of membrane current and its application to conduction and excitation in nerve. *J Physiol.* 117:500-544.

Holmqvist, M.H., J. Cao, R. Hernandez-Pineda, M.D. Jacobson, K.I. Carroll, M.A. Sung, M. Betty, P. Ge, K.J. Gilbride, M.E. Brown, M.E. Jurman, D. Lawson, I. Silos-Santiago, Y. Xie, M. Covarrubias, K.J. Rhodes, P.S. Distefano, and W.F. An. 2002a. Elimination of fast inactivation in Kv4 A-type potassium channels by an auxiliary subunit domain. *Proc. Natl. Acad. Sci. USA*. 99:1035-1040.

Holmqvist, M.H., J. Cao, R. Hernandez-Pineda, M.D. Jacobson, K.I. Carroll, M.A. Sung, M. Betty, P. Ge, K.J. Gilbride, M.E. Brown, M.E. Jurman, D. Lawson, I. Silos-Santiago, Y. Xie, M. Covarrubias, K.J. Rhodes, P.S. Distefano, and W.F. An. 2002b. Elimination of fast inactivation in Kv4 A-type potassium channels by an auxiliary subunit domain. *Proc Natl Acad Sci U S A*. 99:1035-1040.

Holte, R.C. 1993. Very simple classification rules perform well on most commonly used datasets. *Machine Learning*. 11:63-91.

Hong, K.H., and C. Miller. 2000. The lipid-protein interface of a Shaker K(+) channel. *J Gen Physiol.* 115:51-58.

Horn, F., A.L. Lau, and F.E. Cohen. 2004. Automated extraction of mutation data from the literature: application of MuteXt to G protein-coupled receptors and nuclear hormone receptors. *Bioinformatics*. 20:557-568.

Horn, F., G. Vriend, and F.E. Cohen. 2001. Collecting and harvesting biological data: the GPCRDB and NucleaRDB information systems. *Nucleic Acids Res*. 29:346-349.

Horn, F., J. Weare, M.W. Beukers, S. Horsch, A. Bairoch, W. Chen, O. Edvardsen, F. Campagne, and G. Vriend. 1998. GPCRDB: an information system for G protein-coupled receptors. *Nucleic Acids Res*. 26:275-279.

Inza, I., P. Larranaga, R. Blanco, and A.J. Cerrolaza. 2004. Filter versus wrapper gene selection approaches in DNA microarray domains. *Artif. Intell. Med*. 31:91-103.

Isacoff, E.Y., Y.N. Jan, and L.Y. Jan. 1990. Evidence for the formation of heteromultimeric potassium channels in Xenopus oocytes. *Nature*. 345:530-534.

Jain, R., and J. Mazumdar. 2003. A genetic algorithm based nearest neighbor classification to breast cancer diagnosis. *Australas. Phys. Eng. Sci. Med*. 26:6-11.

Jan, L.Y., and Y.N. Jan. 1997a. Cloned potassium channels from eukaryotes and prokaryotes. *Annu. Rev. Neurosci*. 20:91-123.

Jan, L.Y., and Y.N. Jan. 1997b. Voltage-gated and inwardly rectifying potassium channels. *J Physiol*. 505 (Pt 2):267-282.

Jelonek, J., and J. Stefanowski. 1997. Feature subset selection for classification of histological images. *Artif. Intell. Med*. 9:227-239.

Jentsch, T.J. 2000. Neuronal KCNQ potassium channels: physiology and role in disease. *Nature Reviews Neuroscience*. 1:21-30.

Jerez-Aragones, J.M., J.A. Gomez-Ruiz, G. Ramos-Jimenez, J. Munoz-Perez, and E. Alba-Conejo. 2003. A combined neural network and decision trees model for prognosis of breast cancer relapse. *Artif. Intell. Med*. 27:45-63.

Jiang, Y., A. Lee, J. Chen, M. Cadene, B.T. Chait, and R. MacKinnon. 2002a. Crystal structure and mechanism of a calcium-gated potassium channel. *Nature*. 417:515-522.

Jiang, Y., A. Lee, J. Chen, M. Cadene, B.T. Chait, and R. MacKinnon. 2002b. The open pore conformation of potassium channels. *Nature*. 417:523-526.

Jiang, Y., A. Lee, J. Chen, V. Ruta, M. Cadene, B.T. Chait, and R. MacKinnon. 2003a. X-ray structure of a voltage-dependent K+ channel. *Nature*. 423:33-41.

Jiang, Y., A. Pico, M. Cadene, B.T. Chait, and R. MacKinnon. 2001. Structure of the RCK domain from the E. coli K+ channel and demonstration of its presence in the human BK channel. *Neuron*. 29:593-601.

Jiang, Y., V. Ruta, J. Chen, A. Lee, and R. MacKinnon. 2003b. The principle of gating charge movement in a voltage-dependent K+ channel. *Nature*. 423:42-48.

Jolliffe, I.T. 2002. Principal Component Analysis., 2 edition. Springer-Verlag, New York.

Kaneko, S., M. Okada, H. Iwasa, K. Yamakawa, and S. Hirose. 2002. Genetics of epilepsy: current status and perspectives. *Neurosci Res*. 44:11-30.

Kapetanovic, I.M., S. Rosenfeld, and G. Izmirlian. 2004. Overview of commonly used bioinformatics methods and their applications. *Ann N Y Acad Sci*. 1020:10-21.

Kim, H.Y. 2004a. Binary halftone image resolution increasing by decision tree learning. *IEEE Trans Image Process*. 13:1136-1146.

Kim, S. 2004b. Protein beta-turn prediction using nearest-neighbor method. *Bioinformatics*. 20:40-44.

Kira, K., and L.A. Rendell. 1992. The feature selection problem: Traditional methods and a new algorithm. *In* Tenth National Conference on Artificial Intelligence. MIT Press, San Jose, CA. 129-134.

Kohavi, R., and G.H. John. 1997. Wrappers for feature subset selection. *Artificial Intelligence*. 97:273-324.

Koller, D., and M. Sahami. 1996. Toward Optimal Feature Selection. *In* The 13th International Conference on Machine Learning, Bari, Italy.

Koni, P.A., R. Khanna, M.C. Chang, M.D. Tang, L.K. Kaczmarek, L.C. Schlichter, and R.A. Flavella. 2003. Compensatory anion currents in Kv1.3 channel-deficient thymocytes. *J. Biol. Chem*. 278:39443-39451.

Koprowski, P., and A. Kubalski. 2001. Bacterial ion channels and their eukaryotic homologues. *Bioessays*. 23:1148-1158.

Koradi, R., M. Billeter, and K. Wuthrich. 1996. MOLMOL: a program for display and analysis of macromolecular structures. *J Mol Graph*. 14:51-55, 29-32.

Kreusch, A., P.J. Pfaffinger, C.F. Stevens, and S. Choe. 1998. Crystal structure of the tetramerization domain of the Shaker potassium channel. *Nature*. 392:945-948.

Krogh, A., B. Larsson, G. von Heijne, and E.L. Sonnhammer. 2001. Predicting transmembrane protein topology with a hidden Markov model: application to complete genomes. *J Mol Biol*. 305:567-580.

Kuo, A., J.M. Gulbis, J.F. Antcliff, T. Rahman, E.D. Lowe, J. Zimmer, J. Cuthbertson, F.M. Ashcroft, T. Ezaki, and D.A. Doyle. 2003. Crystal structure of the potassium channel KirBac1.1 in the closed state. *Science*. 300:1922-1926.

Laine, M., M.C. Lin, J.P. Bannister, W.R. Silverman, A.F. Mock, B. Roux, and D.M. Papazian. 2003. Atomic proximity between S4 segment and pore domain in Shaker potassium channels. *Neuron*. 39:467-481.

Laine, M., D.M. Papazian, and B. Roux. 2004. Critical assessment of a proposed model of Shaker. *FEBS Lett*. 564:257-263.

Larsson, H.P., O.S. Baker, D.S. Dhillon, and E.Y. Isacoff. 1996. Transmembrane movement of the shaker K+ channel S4. *Neuron*. 16:387-397.

Lerche, H., K. Jurkat-Rott, and F. Lehmann-Horn. 2001. Ion channels and epilepsy. *Am. J. Med. Genet*. 106:146-159.

Li, B., and W.J. Gallin. 2004. VKCDB: Voltage-gated potassium channel database. *BMC Bioinformatics*. 5:3.

Li-Smerin, Y., D.H. Hackos, and K.J. Swartz. 2000. alpha-helical structural elements within the voltage-sensing domains of a K(+) channel. *J. Gen. Physiol*. 115:33-50.

Liman, E.R., P. Hess, F. Weaver, and G. Koren. 1991. Voltage-sensing residues in the S4 region of a mammalian K+ channel. *Nature*. 353:752-756.

Liman, E.R., J. Tytgat, and P. Hess. 1992. Subunit stoichiometry of a mammalian K+ channel determined by construction of multimeric cDNAs. *Neuron*. 9:861-871.

Listgarten, J., S. Damaraju, B. Poulin, L. Cook, J. Dufour, A. Driga, J. Mackey, D. Wishart, R. Greiner, and B. Zanke. 2004. Predictive models for breast cancer susceptibility from multiple single nucleotide polymorphisms. *Clin. Cancer Res*. 10:2725-2737.

Liu, H., and L. Wong. 2003. Data mining tools for biological sequences. *J Bioinform Comput Biol.* 1:139-167.

Liu, H.L., and J.C. Lin. 2004. A set of homology models of pore loop domain of six eukaryotic voltage-gated potassium channels Kv1.1-Kv1.6. *Proteins.* 55:558-567.

Lu, Z., A.M. Klem, and Y. Ramu. 2002. Coupling between voltage sensors and activation gate in voltage-gated K+ channels. *J Gen Physiol.* 120:663-676.

MacKinnon, R. 1991a. Determination of the subunit stoichiometry of a voltage-activated potassium channel. *Nature.* 350:232-235.

MacKinnon, R. 1991b. New insights into the structure and function of potassium channels. *Curr. Opin. Neurobiol.* 1:14-19.

MacKinnon, R. 1991c. Using mutagenesis to study potassium channel mechanisms. *J. Bioenerg. Biomembr.* 23:647-663.

MacKinnon, R. 2004. Potassium channels and the atomic basis of selective ion conduction (Nobel Lecture). *Angew. Chem. Int. Ed. Engl.* 43:4265-4277.

MacKinnon, R., and G. Yellen. 1990. Mutations affecting TEA blockade and ion permeation in voltage-activated K+ channels. *Science.* 250:276-279.

Maiti, R., G.H. Van Domselaar, H. Zhang, and D.S. Wishart. 2004. SuperPose: a simple server for sophisticated structural superposition. *Nucleic Acids Res.* 32:W590-594.

Mannuzzu, L.M., and E.Y. Isacoff. 2000. Independence and cooperativity in rearrangements of a potassium channel voltage sensor revealed by single subunit fluorescence. *J Gen Physiol.* 115:257-268.

Mao, K.Z. 2004. Feature subset selection for support vector machines through discriminative function pruning analysis. *IEEE Trans Syst Man Cybern B Cybern.* 34:60-67.

Masic, N., A. Gagro, S. Rabatic, A. Sabioncello, G. Dasic, B. Jaksic, and B. Vitale. 1998. Decision-tree approach to the immunophenotype-based prognosis of the B-cell chronic lymphocytic leukemia. *Am. J. Hematol.* 59:143-148.

McEntyre, J., and D. Lipman. 2001. PubMed: bridging the information gap. *Cmaj.* 164:1317-1319.

Mendez, M.A., C. Hodar, C. Vulpe, M. Gonzalez, and V. Cambiazo. 2002. Discriminant analysis to evaluate clustering of gene expression data. *FEBS Lett.* 522:24-28.

Mika, S., and B. Rost. 2004. Protein names precisely peeled off free text. *Bioinformatics.* 20 Suppl 1:I241-I247.

Miller, C. 1991. 1990: annus mirabilis of potassium channels. *Science.* 252:1092-1096.

Mingers, J. 1989. An empirical comparison of pruning methods for decision-tree induction. *Machine Learning.* 4:227-243.

Minor, D.L., Y.F. Lin, B.C. Mobley, A. Avelar, Y.N. Jan, L.Y. Jan, and J.M. Berger. 2000. The polar T1 interface is linked to conformational changes that open the voltage-gated potassium channel. *Cell.* 102:657-670.

Mitchell, T.M. 1997. Machine learning. McGraw-Hill, New York, NY. xvii, 414 p. pp.

Monks, S.A., D.J. Needleman, and C. Miller. 1999. Helical structure and packing orientation of the S2 segment in the Shaker K+ channel. *J Gen Physiol.* 113:415-423.

Morais Cabral, J.H., A. Lee, S.L. Cohen, B.T. Chait, M. Li, and R. Mackinnon. 1998. Crystal structure and functional analysis of the HERG potassium channel N terminus: a eukaryotic PAS domain. *Cell.* 95:649-655.

Murphy, C.K. 2001. Identifying diagnostic errors with induced decision trees. *Med. Decis. Making.* 21:368-375.

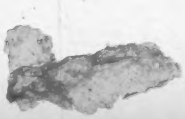
Murrell-Lagnado, R.D., and R.W. Aldrich. 1993. Interactions of amino terminal domains of Shaker K channels with a pore blocking site studied with synthetic peptides. *J Gen Physiol.* 102:949-975.

Narayanan, A., E.C. Keedwell, and B. Olsson. 2002. Artificial intelligence techniques for bioinformatics. *Appl Bioinformatics.* 1:191-222.

Nelson, R.D., G. Kuan, M.H. Saier, Jr., and M. Montal. 1999. Modular assembly of voltage-gated channel proteins: a sequence analysis and phylogenetic study. *J Mol Microbiol Biotechnol.* 1:281-287.

Nguyen, D.V., and D.M. Rocke. 2002. Tumor classification by partial least squares using microarray gene expression data. *Bioinformatics.* 18:39-50.

Noda, M., S. Shimizu, T. Tanabe, T. Takai, T. Kayano, T. Ikeda, H. Takahashi, H. Nakayama, Y. Kanaoka, N. Minamino, and et al. 1984. Primary structure of



Electrophorus electricus sodium channel deduced from cDNA sequence. *Nature*. 312:121-127.

Noskov, S.Y., S. Berneche, and B. Roux. 2004. Control of ion selectivity in potassium channels by electrostatic and dynamic properties of carbonyl ligands. *Nature*. 431:830-834.

Overturf, K.E., S.N. Russell, A. Carl, F. Vogalis, P.J. Hart, J.R. Hume, K.M. Sanders, and B. Horowitz. 1994. Cloning and characterization of a Kv1.5 delayed rectifier K+ channel from vascular and visceral smooth muscles. *Am. J. Physiol.* 267:C1231-1238.

Papazian, D.M., X.M. Shao, S.A. Seoh, A.F. Mock, Y. Huang, and D.H. Wainstock. 1995. Electrostatic interactions of S4 voltage sensor in Shaker K+ channel. *Neuron*. 14:1293-1301.

Papazian, D.M., L.C. Timpe, Y.N. Jan, and L.Y. Jan. 1991. Alteration of voltage-dependence of Shaker potassium channel by mutations in the S4 sequence. *Nature*. 349:305-310.

Patten, C.D., M. Caprini, R. Planells-Cases, and M. Montal. 1999. Structural and functional modularity of voltage-gated potassium channels. *FEBS Lett*. 463:375-381.

Patton, D.E., T. Silva, and F. Bezanilla. 1997. RNA editing generates a diverse array of transcripts encoding squid Kv2 K+ channels with altered functional properties. *Neuron*. 19:711-722.

Potter, D.M. 2004. A permutation test for inference in logistic regression with small- and moderate-sized data sets. *Stat. Med.*

Quinlan, J.R. 1986. Induction of decision trees. *Machine Learning*. 1:81-106.

Quinlan, J.R. 1993. C4.5: Programs for Machine Learning. Morgan Kaufmann, San Mateo, CA.

Rae, J.L., and A.R. Shepard. 2000. Kv3.3 potassium channels in lens epithelium and corneal endothelium. *Exp. Eye Res*. 70:339-348.

Ramaswami, M., M. Gautam, A. Kamb, B. Rudy, M.A. Tanouye, and M.K. Mathew. 1990. Human potassium channel genes: molecular cloning and functional expression. *Mol. Cell. Neurosci*. 1:214-223.

Rettig, J., F. Wunder, M. Stocker, R. Lichtinghagen, F. Mastiaux, S. Beckh, W. Kues, P. Pedarzani, K.H. Schroter, J.P. Ruppersberg, and et al. 1992. Characterization of a Shaw-related potassium channel family in rat brain. *EMBO J.* 11:2473-2486.

Ringner, M., and C. Peterson. 2003. Microarray-based cancer diagnosis with artificial neural networks. *Biotechniques.* Suppl:30-35.

Rost, B., and C. Sander. 1994. Combining evolutionary information and neural networks to predict protein secondary structure. *Proteins.* 19:55-72.

Routledge, R.D. 1997. P-values from permutation and F-tests. *Computational Statistics and Data Analysis.* 24:379-386.

Salvador-Recatala, V., W.J. Gallin, J. Abbruzzese, P.C. Ruben, and A.N. Spencer. 2004. The Structure and Function of a Kv4-like Potassium Channel Expressed in the Myocardium of the Tunicate, *Ciona intestinalis. Submitted.*

Sather, W.A., and E.W. McCleskey. 2003. Permeation and selectivity in calcium channels. *Annu. Rev. Physiol.* 65:133-159.

Scholle, A., R. Koopmann, T. Leicher, J. Ludwig, O. Pongs, and K. Benndorf. 2000. Structural elements determining activation kinetics in Kv2.1. *Receptors Channels.* 7:65-75.

Schrempf, H., O. Schmidt, R. Kummerlen, S. Hinnah, D. Muller, M. Betzler, T. Steinkamp, and R. Wagner. 1995. A prokaryotic potassium ion channel with two predicted transmembrane segments from Streptomyces lividans. *Embo J.* 14:5170-5178.

Schroter, K.H., J.P. Ruppersberg, F. Wunder, J. Rettig, M. Stocker, and O. Pongs. 1991. Cloning and functional expression of a TEA-sensitive A-type potassium channel from rat brain. *FEBS Lett.* 278:211-216.

Schwartz, R.M., and M.O. Dayhoff. 1978. Matrices for detecting distant relationships. *In* Atlas of Protein Sequence and Structure. Vol. 5, suppl. 3. M.O. Dayhoff, editor. Natl. Biolmed. Res. Found., Washington, D.C. 345-352.

Selbig, J., T. Mevissen, and T. Lengauer. 1999. Decision tree-based formation of consensus protein secondary structure prediction. *Bioinformatics.* 15:1039-1046.

Shen, N.V., X. Chen, M.M. Boyer, and P.J. Pfaffinger. 1993. Deletion analysis of K+ channel assembly. *Neuron.* 11:67-76.

Sigworth, F.J. 1994. Voltage gating of ion channels. *Q. Rev. Biophys.* 27:1-40.

Smits, P. 1996. Role of potassium channels in the modulation of insulin release. *Diabetologia.* 39:865-867.

Sokolova, O., L. Kolmakova-Partensky, and N. Grigorieff. 2001. Three-dimensional structure of a voltage-gated potassium channel at 2.5 nm resolution. *Structure.* 9:215-220.

Stanfield, P.R., S. Nakajima, and Y. Nakajima. 2002. Constitutively active and G-protein coupled inward rectifier K+ channels: Kir2.0 and Kir3.0. *Rev Physiol Biochem Pharmacol.* 145:47-179.

Starace, D.M., E. Stefani, and F. Bezanilla. 1997. Voltage-dependent proton transport by the voltage sensor of the Shaker K+ channel. *Neuron.* 19:1319-1327.

Stuhmer, W., J.P. Ruppersberg, K.H. Schroter, B. Sakmann, M. Stocker, K.P. Giese, A. Perschke, A. Baumann, and O. Pongs. 1989. Molecular basis of functional diversity of voltage-gated potassium channels in mammalian brain. *EMBO J.* 8:3235-3244.

Tag, P.M., and J.E. Peak. 1996. Machine learning of maritime fog forecast rules. *J Appl Meteorol.* 35:714-724.

Tempel, B.L., D.M. Papazian, T.L. Schwarz, Y.N. Jan, and L.Y. Jan. 1987. Sequence of a probable potassium channel component encoded at Shaker locus of Drosophila. *Science.* 237:770-775.

Thompson, J.D., D.G. Higgins, and T.J. Gibson. 1994. CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res.* 22:4673-4680.

Tiwari-Woodruff, S.K., C.T. Schulteis, A.F. Mock, and D.M. Papazian. 1997. Electrostatic interactions between transmembrane segments mediate folding of Shaker K+ channel subunits. *Biophys J.* 72:1489-1500.

Towbin, J.A., and M. Vatta. 2001. Molecular biology and the prolonged QT syndromes. *Am J Med.* 110:385-398.

Treptow, W., B. Maigret, C. Chipot, and M. Tarek. 2004. Coupled Motions between Pore and Voltage-Sensor Domains: A Model for Shaker B, a Voltage-Gated Potassium Channel. *Biophys J.* 87:2365-2379.

Tsai, W., A.D. Morielli, T.G. Cachero, and E.G. Peralta. 1999. Receptor protein tyrosine phosphatase alpha participates in the m1 muscarinic acetylcholine receptor-dependent regulation of Kv1.2 channel activity. *Embo J.* 18:109-118.

Vapnik, V.N. 1995. The Nature of Statistical Learning Theory. Springer-Verlag, Berlin.

Viikki, K., E. Kentala, M. Juhola, I. Pyykko, and P. Honkavaara. 2002. Generating decision trees from otoneurological data with a variable grouping method. *J. Med. Syst.* 26:415-425.

Vogalis, F., M. Ward, and B. Horowitz. 1995. Suppression of two cloned smooth muscle-derived delayed rectifier potassium channels by cholinergic agonists and phorbol esters. *Mol Pharmacol.* 48:1015-1023.

Weber, G., S. Vinterbo, and L. Ohno-Machado. 2004. Multivariate selection of genetic markers in diagnostic classification. *Artif. Intell. Med.* 31:155-167.

Weiss, J.L., J. Yang, C. Jie, D.L. Walker, S. Ahmed, Y. Zhu, Y. Huang, K.M. Johansen, and J. Johansen. 1999. Molecular cloning and characterization of LKv1, a novel voltage-gated potassium channel in leech. *J. Neurobiol.* 38:287-299.

Willard, L., A. Ranjan, H. Zhang, H. Monzavi, R.F. Boyko, B.D. Sykes, and D.S. Wishart. 2003. VADAR: a web server for quantitative evaluation of protein structure quality. *Nucleic Acids Res.* 31:3316-3319.

Williamson, I.M., S.J. Alvis, J.M. East, and A.G. Lee. 2003. The potassium channel KcsA and its interaction with the lipid bilayer. *Cell Mol Life Sci.* 60:1581-1590.

Wishart, D.S., R.F. Boyko, and B.D. Sykes. 1994. Constrained multiple sequence alignment using XALIGN. *Comput. Appl. Biosci.* 10:687-688.

Witten, I.H., and E. Frank. 2000. Data mining : practical machine learning tools and techniques with Java implementations. Morgan Kaufmann, San Francisco, CA. xxv, 371 p. pp.

Wollmuth, L.P., and A.I. Sobolevsky. 2004. Structure and gating of the glutamate receptor ion channel. *Trends Neurosci.* 27:321-328.

Wood, J.N., and M. Baker. 2001. Voltage-gated sodium channels. *Curr Opin Pharmacol.* 1:17-21.

Wu, S.N. 2003. Large-conductance Ca2+- activated K+ channels:physiological role and pharmacology. *Curr Med Chem.* 10:649-661.

Yan, C., D. Dobbs, and V. Honavar. 2004. A two-stage classifier for identification of protein-protein interface residues. *Bioinformatics.* 20 Suppl 1:I371-I378.

Yang, N., A.L. George, Jr., and R. Horn. 1996. Molecular basis of charge movement in voltage-gated sodium channels. *Neuron.* 16:113-122.

Yang, N., and R. Horn. 1995. Evidence for voltage-dependent S4 movement in sodium channels. *Neuron.* 15:213-218.

Yang, Y., and J.O. Pedersen. 1997. A Comparative Study on Feature Selection in Text Categorization. *In* The 14th International Conference on Machine Learning, Nashville, TN.

Yellen, G. 1998. The moving parts of voltage-gated ion channels. *Q Rev Biophys.* 31:239-295.

Yellen, G. 2001. Keeping K+ completely comfortable. *Nat Struct Biol.* 8:1011-1013.

Yellen, G. 2002. The voltage-gated potassium channels and their relatives. *Nature.* 419:35-42.

Yi, T.M., and E.S. Lander. 1993. Protein secondary structure prediction using nearest-neighbor methods. *J. Mol. Biol.* 232:1117-1129.

Yifrach, O., and R. MacKinnon. 2002. Energetics of pore opening in a voltage-gated K(+) channel. *Cell.* 111:231-239.

Zagotta, W.N., and S.A. Siegelbaum. 1996. Structure and function of cyclic nucleotide-gated channels. *Annu. Rev. Neurosci.* 19:235-263.

Zhao, L.P., R. Prentice, and L. Breeden. 2001. Statistical modeling of large microarray data sets to identify stimulus-response profiles. *Proc Natl Acad Sci U S A.* 98:5631-5636.

Zhou, W., Y. Qian, K. Kunjilwar, P.J. Pfaffinger, and S. Choe. 2004. Structural insights into the functional interaction of KChIP1 with Shal-type K(+) channels. *Neuron.* 41:573-586.

Zorman, M., H.P. Eich, B. Stiglic, C. Ohmann, and M. Lenic. 2002. Does size really matter--using a decision tree approach for comparison of three different databases from the medical field of acute appendicitis. *J. Med. Syst.* 26:465-477.